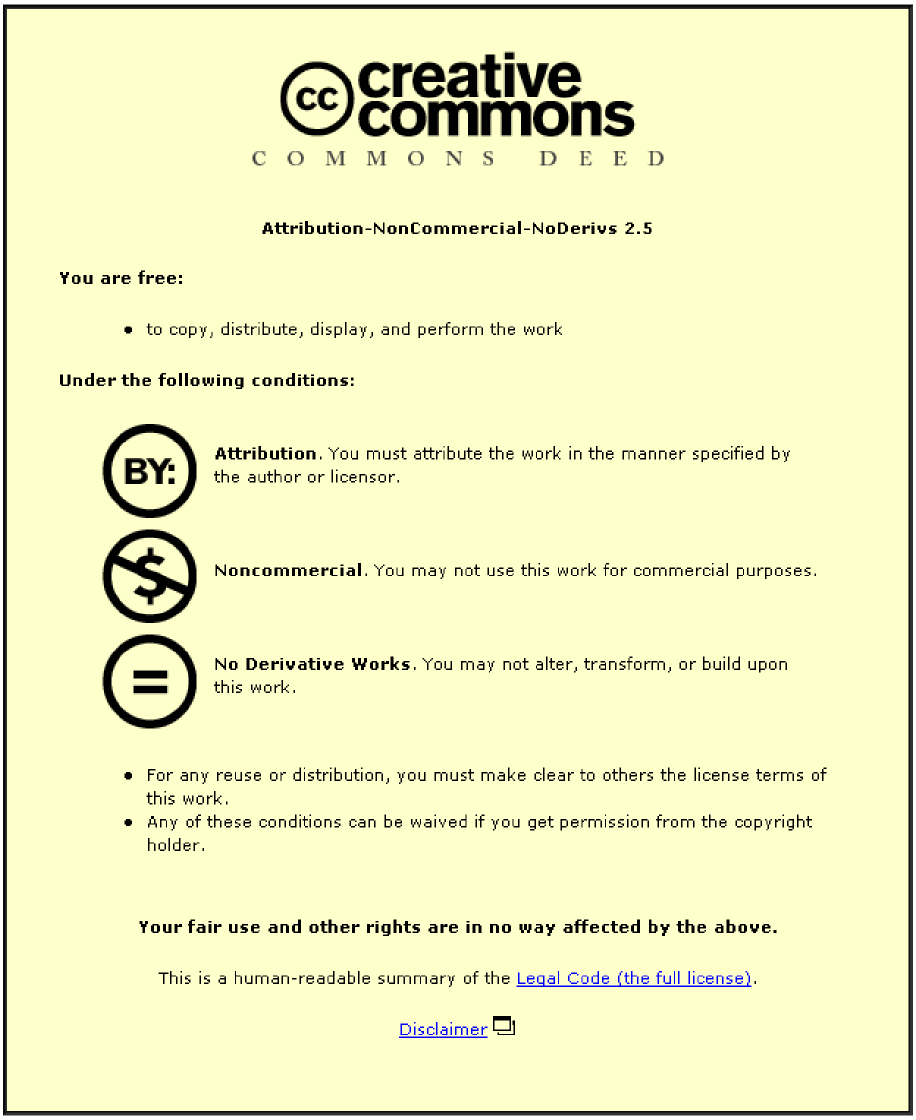


Loughborough
University

Thesis Access Form

Copy No……….……………………**Location**……………………………………..……………...…

Author…Muhammad Baqar Raza……..…….

Title… A Knowledge Based Approach to Integration of Products, Processes and Reconfigurable Automation Resources **..**

Status of access OPEN / **RESTRICTED** / CONFIDENTIAL

Moratorium Period: ..... 05 ……..……………years, ending……April……. /……2017…………….

Conditions of access approved by (CAPITALS):…Prof. Robert Harrison........…

Supervisor (Signature)………Prof. Robert Harrison…………………..

School of…Mechanical and Manufacturing Engineering………

Author's Declaration: *I agree the following conditions:*

Open access work shall be made available (in the University and externally) and reproduced as necessary at the discretion of the University Librarian or Dean of School. It may also be digitised by the British Library and made freely available on the Internet to registered users of the EThOS service subject to the EThOS supply agreements.

*The statement itself shall apply to **ALL** copies including electronic copies:*

This copy has been supplied on the understanding that it is copyright material and that no quotation from the thesis may be published without proper acknowledgement.

Restricted/confidential work: All access and any photocopying shall be strictly subject to written permission from the University Dean of School and any external sponsor, if any.

Author's signature …...Muhammad Baqar Raza…......... Date…………23 April, 2012…...……..

<table>
<tr><td colspan="4"><b>users declaration:</b> for signature during any Moratorium period (Not Open work):
<b><i>I undertake to uphold the above conditions:</i></b></td></tr>
<tr><td>Date</td><td>Name (CAPITALS)</td><td>Signature</td><td>Address</td></tr>
<tr><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td></tr>
</table>

A Knowledge Based Approach to Integration of Products, Processes and Reconfigurable Automation Resources

By

Muhammad Baqar Raza

Doctoral Thesis

Submitted in partial fulfilment of the requirements for the award of

Doctor of Philosophy

of Loughborough University

April, 2012

Manufacturing Systems Integration Research Institute

Loughborough University

CERTIFICATE OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this thesis. That the original work is my own except as specified in acknowledgements or in footnotes and that neither the thesis nor the original work contained therein has been submitted to this or any other institution for a degree.

… Muhammad Baqar Raza …………. (Signed)

……23 April, 2012 ………………………. (Date)

DEDICATION

Dedicated To My Parents

Imagination is more important than knowledge. For knowledge is limited to all we now know and understand, while imagination embraces the entire world, and all there ever will be known and understood. The logic will get you from A to B, imagination will take you everywhere.

Albert Einstein (1879 - 1955)

ACKNOWLEDGEMENTS

There are no words eloquent enough to convey my sincere gratitude towards my research supervisor Prof. Robert Harrison for his continuous guidance and serene supervision throughout my PhD study. This particular research would not have been possible without his constant help and support in all forms. His comments have always been apt, suggestions encouraging, discussions inspiring and support relentless. I have had the freedom to have numerous invaluable discussions, thoughts sharing, and suggestions which have shaped my analytical and reasoning ability and kept me motivated all along the journey and I genuinely consider being honoured working with him.

I would also like to thank the other members of staff in the school who have all at one time or another been helpful, encouraging or just plain interested.

My appreciation also goes to other colleagues and friends with whom I have discussed many of the ideas that are in this thesis. Dr. Thomas Kirkham, who later joined Nottingham University, yet remained morally helpful and supportive. I am very much thankful to Dr. Atul Jain for his support and especially playing the part of a bridge between Ford Company and MSI Research Institute of Loughborough University. I kept chasing him for discussions and refinement of the research work as well as for his impartial comments on the developed system, especially long discussions on validation of the developed knowledge based system. Also I am grateful to Stuart McLeod, Dr. Radmehr Monfared and Dr. Daniel Vera from the same research group for their kind support and guidance, when required. I would also like to thank Senior Lecturer Bob Young and Dr. George Gunendran from Wolfson School for helpful discussions especially for evaluating the ‘e2ks’ KM tool being used at Ford. The CECA (Centre of Excellence in Customised Assembly, Loughborough University) team members, Rob, Joe and later Andy for providing previous research results. Alison Howells and Samantha Finney from Research Student Office and Louise Waldron and Martin Matthews from Student Advice and Support Service remained extremely helpful and their presence has made the journey easier.

In addition to them, I’m also thankful to Ford engineers Andy Baker, Paul and Roger from Dunton Technical Centre. Most importantly, I am obliged to Program Manager Powertrains Dr. Leslie Lee for sharing his tons of experience and shaping the outcome of this research. His reasoning views directly influenced my curiosity and stimulated dormant portions of my brain.

I have had the opportunity to meet Dr. Sanaullah Nazir from DERI, Galway, Ireland and am indebted to him for introducing me a robust ontology language (WSML) as well as introducing the Java APIs specifically required for the WSMT IDE, used in the current research.

A special thanks goes to the external and internal examiners of this thesis. I'm grateful to Prof. Andy West and Prof. Philip Moore for their comments and suggestions regarding improvements in this thesis and research study.

I feel a deep sense of appreciation to my parents for their endless love that has guided all my visions and formed the most important part of growing-up. They have constantly been praying for my success back home and their moral support and good wishes have been the asset of my life. Thanks to my elder sister and her family for all the care and support when required. Thanks also to my little nephews for cheering me up out of occasional frustrations.

This research was funded by Business Driven Automation Project and Ford Motor Company through IMCRC. I am grateful to IMCRC and the Loughborough University, and in turn, EPSRC, UK, for providing me the opportunity to carry out the current research.

ABSTRACT

The success of next generation automotive companies will depend upon their ability to adapt to ever changing market trends thus becoming highly responsive. In the automotive sector, the assembly line design and reconfiguration is an especially critical and extremely complex job. The current research addresses some of the aspects of this activity under the umbrella of a larger ongoing research project called Business Driven Automation (BDA) project. The BDA project aims to carry out complete virtual 3D modeling-based verifications of the assembly line for new or revised products in contrast to the prevalent practice of manual evaluation of effects of product change on physical resources. The overall project has already contributed to successfully build and implement a modular, lightweight, modeling and simulation tool for the rapid design and reconfiguration of assembly automation systems. The said tool called ‘Core Component Editor’ (CCE) is based upon Component Based (CB) technology utilising pre-defined set of tested and reusable mechanisms. These mechanisms can be cloned and assembled to make new automation systems for the changed product. However, there are two limitations associated with the current CCE tool, (i) though the CCE tool has significantly reduced the simulation time, it is still a labour intensive task to carry out complete virtual verification of the entire assembly line for every change in the product, and (ii) the CCE tool is primarily a modeling and simulation tool, it lacks knowledge capture and management capabilities. Therefore a new approach is proposed which can replace the current practice of repeated simulation analyses performed against frequent product changes by recording the simulation results in a reusable form and capturing the once-performed simulation analysis results for subsequent product changes. This research explores the possibility of developing and using such an approach for rapid parameterised analysis of Product, Process and Resource (PPR) inter-dependency constraints. The research makes a case for the faster evaluation of product-resource constraints using knowledge engineering principles and ontology representation formalism. The research successfully designed and developed a knowledge based system to facilitate creating a knowledge reuse infrastructure in the domain of automation engine assembly.

Key Words: Knowledge based system, knowledge engineering, ontology engineering, reconfigurable manufacturing, powertrain assembly line, engine fit, product process resource relations.

LIST OF PUBLICATIONS

Although this thesis is written in the form of a monograph, some parts of the contents have been published as papers in refereed scientific journals and peer reviewed conference proceedings. The publications are listed below in Table 1. See also chapter 8, section 8.1.

	Fully Refereed Journal Publications
	Published
Paper 1	Raza Muhammad Baqar, Robert Harrison, Tom Kirkham, "*Knowledge Based Product Lifecycle Management System*", International Journal of Computing Architecture, Volume 4, Issue 2, July 2009
Paper 2	Raza Muhammad Baqar, Robert Harrison, Tom Kirkham, "*Efficient Knowledge Management through PLM Systems*", Journal of E Technology, Volume 1, Issue 2, May 2010, pp 66-74.
Paper 3	Raza Muhammad Baqar, Robert Harrison, Tom Kirkham, Q Reul, "*Knowledge Based Flexible and Integrated PLM System at Ford*", in IJISM , Vol. 1, No.1, March, 2011, pp 8-16.
Paper 4	Raza Muhammad Baqar, Robert Harrison, "*Information Modeling and Knowledge Management Approach to Reconfiguring Manufacturing Enterprises*", IJAIT, World Scientific, Vol. 20 No. 3, June, 2011.
Paper 5	Raza Muhammad Baqar, Robert Harrison, "*Design, Development & Implementation of Ontological Knowledge Based System for Automotive Assembly Lines*", International Journal of Data Mining & Knowledge Management Process (IJDKP) Vol. 1, No.5, September, 2011.
Paper 6	Raza Muhammad Baqar, Robert Harrison, "*Amalgamation of Ontology and Knowledge Engineering in Automotive Industry*", International Journal of Advanced Computer Science and Applications, IJACSA, Vol. 3, No. 1, January, 2012.
Paper 7	Raza Muhammad Baqar, Robert Harrison, Tom Kirkham, "*Embracing Knowledge Driven Devices: Service Oriented Standardisation of Production Management in Automated Lines*", International Journal of Service and Computing Oriented Manufacturing, IJSCOM, Inderscience Publishers, March, 2012 (accepted, in press).
Paper 8	Tom Kirkham, Raza Muhammad Baqar, Sandra Winfield, Robert Harrison, "*Business Process Management and the Challenge of Dynamic Services*", International Journal for Digital Society (IJDS), Published by Infonomics Society, Volume 2, Issue 3, 2011.
	In Press
Paper 9	Raza Muhammad Baqar, Robert Harrison, "*Semantic Annotation for Knowledge Management*": A Corporate Semantic Web, IJMIT.
	Book Sections
Chapter	Raza Muhammad Baqar, Robert Harrison, "*Knowledge-Intensive and Integrated*

	Approach to Problem Solving in Powertrain Assembly Line Design", Information Technology in Management, edited by Prof. Dr. V. Tewari, (ISBN 978-955-0775-00-2)
	Peer Reviewed Conference Papers
Paper 1	Raza Muhammad Baqar, Robert Harrison, Tom Kirkham, "*Evolving Knowledge Based Product Lifecycle Management from a Digital Ecosystem to Support Automated Manufacturing*", Proceedings of the International ACM Conference, MEDES , Lyon, France, October, 2009.
Paper 2	Raza Muhammad Baqar, Robert Harrison, "*Improving Manufacturing Efficiency at Ford using Product Centred Knowledge Management*" in the Proceedings of the IEEE, 4th International Conference on ICDIM, Michigan, USA, November, 2009.
Paper 3	Raza Muhammad Baqar, Robert Harrison, "*Ontological Knowledge Based System for Product, Process and Resource Relationships in Automotive Industry*", Proceedings of the 1st International Conference on Ontology and Semantic Web for Manufacturing (OSEMA, 2011), collocated with the 8th Extended Semantic Web Conference (ESWC), Greece, May, 2011.
Paper 4	Tom Kirkham, Raza Muhammad Baqar, Sandra Winfield, "*Business Process Management and the Challenge of Dynamic Services*", Proceedings of the International Conference on Information Society, i-Society, London, UK, June, 2011.
Paper 5	Raza Muhammad Baqar, Robert Harrison, "*Intelligent Integration of Product Design to Production Devices*", Proceedings of the 10th International Conference on Information and Knowledge Engineering (ICIKE, WORLDCOMP 2011), Las Vegas, Nevada, USA, July, 2011.
Paper 6	Haq I., Raza Muhammad Baqar, Bilal Ahmad, Robert Harrison, "*Product to Process Lifecycle Management in Assembly Automation Systems*", Proceedings of DET2011, 7th International Conference on Digital Enterprise Technology, Athens, Greece, September, 2011.
Paper 7	Raza Muhammad Baqar, Robert Harrison, Tom Kirkham, "*Knowledge Based Services for Devices in Automation*", - Proceedings of the International Joint Conference on Knowledge Engineering and Ontology Development, KEOD, IC3K, Paris, France, October, 2011.
	In Press
Paper 8	Raza Muhammad Baqar, Robert Harrison, "Application of Knowledge Based Engineering Approach to Rapid Design, Reconfiguration and Analysis of Automotive Assembly Lines", Proceedings of the 3rd International Asia Conference on Informatics in Control, Automation and Robotics (CAR 2011), Shenzhen, China, December, 2011.

Table 1: List of Publications

TABLE OF CONTENTS

LIST OF FIGURES

REFERENCE

LIST OF TABLES

REFERENCE

LIST OF ABBREVIATIONS

3D	3 Dimensional
AESOP	ArchitecturE for Service-Oriented Process monitoring and control
AHP	Analytical Hierarchical Process
AI	Artificial Intelligence
ALDIMS	Assembly Line Digital Information Management Systems
API	Application Programming Interface
BDA	Business Driven Automation
BoM	Bill of Material
BoP	Bill of Process
BoR	Bill of Resource
BPEL	Business Process Execution Language
CAAPP	Computer-Aided Assembly Process Planning
CAD	Computer Aided Design
CB	Component Based
CAD	Computer Aided Design
CAE	Computer Aided Engineering
CAM	Computer Aided Manufacturing
CAPP	Computer Aided Process Planning
CBR	Case-based reasoning
CCE	Core Component Editor
CDSS	Collaborative Decision Support Systems
CE	Concurrent Engineering
CIM	Computer Integrated Manufacturing
CoP	Communities of Practices
CPD	Collaborative Product Development
CPDM	Collaborative Product Development Management
CRM	Customer Relationship Management
CSCW	Computer Supported Collaborative Work
CW	ClockWise
CCW	Counter ClockWise
DBC	Database Connectivity
DBMS	Database Management System
DELMIA	Digital Enterprise Lean Manufacturing Interactive Application
DES	Discrete Event Simulation
DfA	Design for Assembly
DfM	Design for Manufacturing
DL	Description Logic
DLMS	Direct Labour Management System
DML	Dedicated Manufacturing Lines
DPA	Digital Pre-Assembly
DSS	Decision Support System
DTC	Dunton Technical Centre

DVP&R	Design Validation Plan and Report
EDMS	Electronic Document Management Software
EI	Enterprise Integration
E2KS	Enterprise Engineering Knowledge System
EKM	Engineering Knowledge Management
ERP	Enterprise Resource Planning
FMS	Flexible Manufacturing Systems
FoF	Factories of Future
FOM	Function Oriented Modularity
GPDS	Global Product Development System
GSPAS	Global Study Process Allocation System
GUI	Graphical User Interface
HTTP	HyperText Transport / Transfer Protocol
HVA	High Value Added
ICT	Information and Communication Technology
IDE	Integrated Development Environment
IPD	Integrated Product Development
IRIS	Integrated Rule Inference System
JSP	Java Server Pages
KB	Knowledge Base / Knowledge Based
KBE	Knowledge Based Engineering
KBS	Knowledge Based System
KC	Key Characteristic
KE	Knowledge Engineering
KM	Knowledge Management
KMS	Knowledge Management System
KOMPRESSA	Knowledge-Oriented Methodology for the Planning and Rapid Engineering of Small-Scale Applications
K-Pac	Knowledge Packet
KR	Knowledge Representation
LP	Logic Programming
LAN	Local Area Network
MAS	Multi Agent System
MDS	Manufacturing Design Specifications
ME	Manufacturing Enterprises
MFD	Modular Function Deployment
MIKE	Model-based and Incremental Knowledge Engineering
MODAPTS	MODular Arrangement of Predetermined Time Standards
MOKA	Methodology & tools Oriented to Knowledge-based Applications
MPS	Manufacturing Process Specification
MPS	Modular Production System
MSE	Manufacturing Structure Editor
MSI	Manufacturing Systems Integration
ODBC	Open Database Connectivity
OEM	Original Equipment Manufacturer
OKBS	Ontological Knowledge Based System

OMT	Object Modeling Technique,
ONTOMAS	ONTOlogy for design of Modular Assembly Systems
OWL	Web Ontology Language
PD	Process Designer
PLC	Programmable Logic Controller
PLM	Product Lifecycle Management
PM	Program Management
PPR	Product Process Resource
PSC	Program Strategy Confirmed
PSL	Process Specification Language
RAS	Reconfigurable Assembly System
R&D	Research and Development
RDF	Resource Description Framework
RDFS	RDF Schema
RMS	Reconfigurable Manufacturing System
RMT	Reconfigurable Machine Tool
RTV	Room Temperature Vulcanising
SAP	Systems Applications and Products
SAS	Smart Assembly System
SBE	Slow Build Event
SE	Simultaneous Engineering
SKOS	Simple Knowledge Organization System
SL	Standard Language
SOA	Service Oriented Architecture
SPARQL	Simple Protocol and RDF Query Language
SWRL	Semantic Web Rule Language
SWS	Semantic Web Service
TC	Teamcenter
TPS	Toyota Production System
UML	Unified Modeling Language
URI	Unified Resource Identifier
URL	Uniform Resource Locator
VBE	Virtual Build Event
VC	Virtual Commissioning
VDS	Virtual Design Studio
VP	Verification Prototype
VR	Virtual Reality
VRML	Virtual Reality Modeling Language
VV	Virtual Verification
WSML	Web Service Modeling Language
WSMT	Web Service Modeling Toolkit
WWW	World Wide Web
XML	eXtensible Mark-up Language
XMI	XML Metadata Interchange

CHAPTER 1. INTRODUCTION

1.1 Background

Manufacturing enterprises, in general, and automotive industries, in particular, are facing enormous pressures in the current dynamic and uncertain business environments. There is an ever growing globalisation, shorter product lifecycles, international business competition, constant technological innovations and stringent environmental regulations posing greater challenges to manufacturing and especially automotive industries. The automotive sector has had a history of providing revolutionary manufacturing paradigms in the last century e.g. Ford's mass production system and Toyota's lean production system. New strategies are still required for the manufacturing / assembly systems in automotive sector to meet new business trends for speed and responsiveness (Aberdeen Group Report, 2008). Recently, a new approach called Modular Production System (MPS) has emerged from research and migrated to industry. The MPS approach is based upon Component Based (CB) technology. The CB technology defines systems into basic building blocks of independent functional units called 'components'. The MPS strategy aims to achieve reconfigurability. The reconfigurability is the ability to repeatedly, quickly and economically change and rearrange the components of a system to perform a different function. MPS helps achieve reuse and reconfigurability and, unlike Flexible Manufacturing Systems (FMS), which require a general purpose high investment system, modular systems are usually dedicated and designed and used for mass customisation. Therefore, a combination of a suitable reconfiguration strategy based upon a modular platform is a potential breakthrough technology in the current volatile environment.

Ford Motor Company is among the world's five largest automobile manufacturers (KPMG Report, 2010) with globally distributed plants and supply chain partners. Loughborough University, working in close collaboration with Ford Motor Company UK, aims to lay down the foundations of a new automation strategy to help utilise reconfiguration based on CB technology. Existing infrastructures do not provide for the rapid design and development of automation resources and it is evident that new tools based on CB technology can contribute to the efficient design of assembly machines.

This research study constitutes part of a larger research programme being carried out under the umbrella of a project called "Business Driven Automation" (BDA[1]) at the MSI Research Institute in Loughborough University. The motivation for this project is a comparative analysis of business needs versus technology options to drive the future design and implementation of automation systems. The project is a collaboration between Loughborough University researchers and industrial partners including Ford Motor Company, ThyssenKrupp Krause GmbH, Bosch-Rexroth and Schneider Electric. The overall aim of the wider research is to introduce modular engineering techniques into the automotive industry. Current automation systems are difficult to service, reconfigure, integrate, and optimise (Harrison et al, 2009). Moving away from traditional, slower practices, the project has taken a modular approach to the physical components and developed a new type of software engineering tool to support the rapid configuration and modification of these factory automation systems[1]. The BDA project looked into the existing shortcomings and designed an application called Core Component Editor (CCE) based upon Component Based (CB) technology for modular design of assembly machines with the goal of reducing the overall time for the virtual design and assembly verification process. The current research goes one step further to more fully exploit the CB approach in terms of knowledge exploitation and reuse.

The development of powertrain (products), assembly processes (processes) and powertrain assembly automation machines (resources) are very complex tasks. Product, Process and Resource (PPR) are the key elements of engineering domain in any automotive industry (Sundin, 2004; Lohse, 2006; Raza et al, 2009a; Jarvenpaa et al, 2010; Lanz, 2010; Raza and Harrison, 2011a; Raza and Harrison, 2011b). In this thesis, the term ‘product’ means assemblies or sub-assemblies of an engine, ‘process’ means process design, planning and implementation to assemble engine, while ‘resource’ means machines / workstations (automatic, semi-automatic or manual) and manpower used on powertrain assembly lines.

The author’s research focuses on the development of approaches for the management of information accumulated within assembly line design and reconfiguration activities. The thesis particularly aims at design and development of ontology based knowledge support for information processing and knowledge management within the automotive industry. As a result, a knowledge based system is designed, developed and test implemented for integration of product, process and resource domains in engineering sector of automotive organisation.

[1] http://www.lboro.ac.uk/departments/mm/research/manufacturing-systems/dsg/bda/index.htm

1.2 Problem Description

Current automotive plant configurations demand capabilities for mixed-product assembly lines in order to assemble many variants of the product with changed layout / reconfigurations of the existing assembly line. Contemporary powertrain assembly lines may consist of up to 150 workstations to cater for the assembly of a variety of similar products. The multi-function and multi-configuration capabilities of the assembly machines are not explicitly known to the product designers and process planners. The virtual verification of new / changed resources against a new / changed product is usually not carried out for all the workstations, and sometimes is not feasible, as it is an extremely time consuming and specialised activity. Similarly, the results of the virtual verifications of the selected machines are not recorded in a reusable format, therefore, the outcome and the decision logic is lost. The changes required in operational parameters of automation machines against the changed product are manually checked because there is no readily available information as to how the changes in the product may affect the corresponding machines. Therefore, a complete confirmation of product-resource constraints for the entire assembly line is rarely achieved. This, however, leads to unexpected problems at later stages of machine build and commissioning which may pose costly engineering changes in Bill of Process (BoP) or Bill of Resource (BoR) or both. The unavailability of the processes' design logic and in turn, resource capabilities and constraints at the conceptual phase of the product design is a major discrepancy which results in target delays at later stages of program management (Raza & Harrison, 2011b).

To support the activities of engineering partners during different phases of lifecycle, mostly ad-hoc integration methods and mechanisms are currently being employed in automotive industries (Molina et al 2005; Harrison et al, 2006). Despite recent advancements in technologies and supported applications, there exist significant deficiencies in system performance especially from the perspectives of knowledge sharing and reusability. Products, processes and resources are designed and built by different teams across the globe and different software applications are used for the design, development and assembly of PPRs. The data structures of these software applications are localised and also evolve over time, changes in data sources are not unusual and the flow of information among functional domains typically increases with time making it difficult to extract information from a single source. Both engines and machines to assemble engines are built in separate applications with separate methodologies, thus integration of these software application is not only a challenge but, is sometimes not possible. In a large scale company such as Ford there can be hundreds of different software applications within the engineering domain. Again different data formats and application systems exist within the Ford and its supply partners which results in rare integration of applications. Although there exist several national and international standards for exchange of data among different software applications, yet an unambiguous standardised interoperability infrastructure is still missing (Ray and Jones, 2006; Lanz, 2010). The software applications are localised and data-centric with emphasis on information management rather than knowledge management. This emphasises the need for standardised information modeling, interoperability infrastructure and knowledge management techniques in the engineering domain. Therefore, efficient design and build of powertrain assembly machines remains a challenge, handicapped by the above problems.

Thus, the author's research has focussed two major problems being faced in the automotive sector, (i) the PPR inter-dependency constraints are currently not explicitly defined thus, product assemblability has to be confirmed through slow manual activities against product alterations, and (ii) PPR relations are scattered around several disparate software applications resulting in a lack of an integrative environment.

1.2.1. Justification for Research

Current automation systems fail to meet business requirements (Raza and Harrison, 2011c; Raza and Harrison, 2011d). The lifecycle support for automation resources does not sufficiently provide capabilities for rapid change and flexible reconfigurations and therefore, are not appropriate when viewed in the flexible business context of today's manufacturing systems (Sharma, 2005). In addition, assembly cost often accounts for over 40 per cent of the total manufacturing cost (Owen, 1985; Li and Hwang, 1992; Venkatachalam et al, 1993; Zha et al, 2001). Therefore, it is essential to take into consideration all the requirements of assembly during the early design stages, otherwise additional cost and time to redesign already finished designs is inevitable (Shehab and Abdalla, 2006).

Competitive advantage today often depends upon the efficient and effective application of the information technology and knowledge engineering (Nonaka and Teece, 2001). Turban et al (2007) has ranked 'making clear cut decisions when needed' as the most important management practice. Further to the discussion, Turban et al (2007) argue that processing information manually while making decisions is becoming increasingly difficult due to several factors including accessibility to global markets, time constraints, sophisticated analysis, increased number of alternatives etc. The biggest loss to a company in the current IT revolutionised world is the loss in knowledge and often occurs due to inadequate day to day communications within team members and especially interactions across different disciplines of manufacturing / assembly as depicted in Figure 1. The knowledge is a linear function with respect to time.

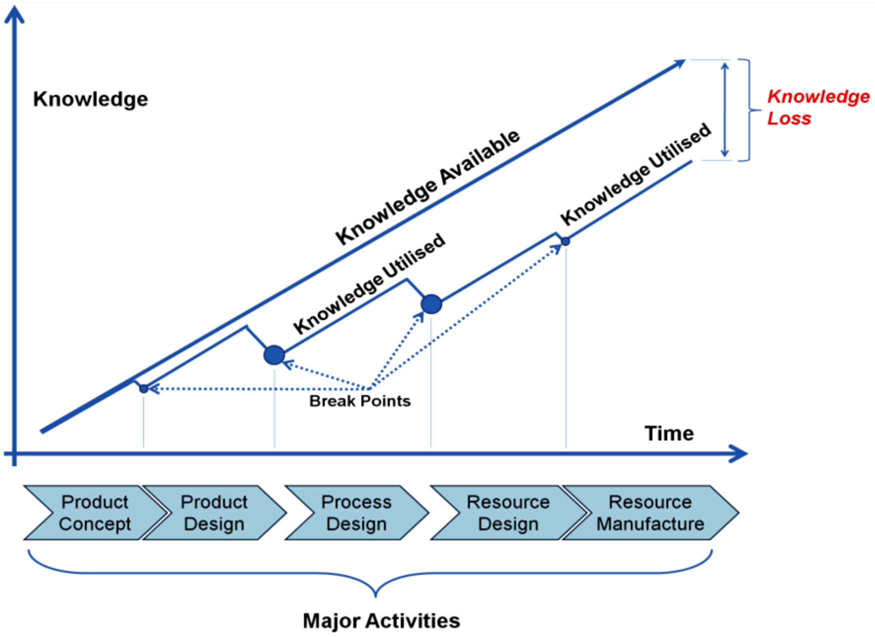


Figure 1: Knowledge loss during communication in disparate domains adopted from (Transparency project[2], 2011)

During the course of inter-departmental or inter-organisational communications and interactions, some knowledge is lost. The knowledge utilised is always a fraction of knowledge available. Present day organisations aim to reduce knowledge loss to a minimum by developing a knowledge retention infrastructure.

Similarly, there is no 'formal and explicit' definition of PPR domains and their inter-relationships in the automotive sector. There is huge potential for the reuse of existing information for the PPR domains and as a rule of thumb, product change typically requires 20% of new resources, 40% modified resources and 40% unchanged resources (Chung et al, 2005; Raza and Harrison, 2011d) as described in Figure 2.

[2] www.transparency-project.eu

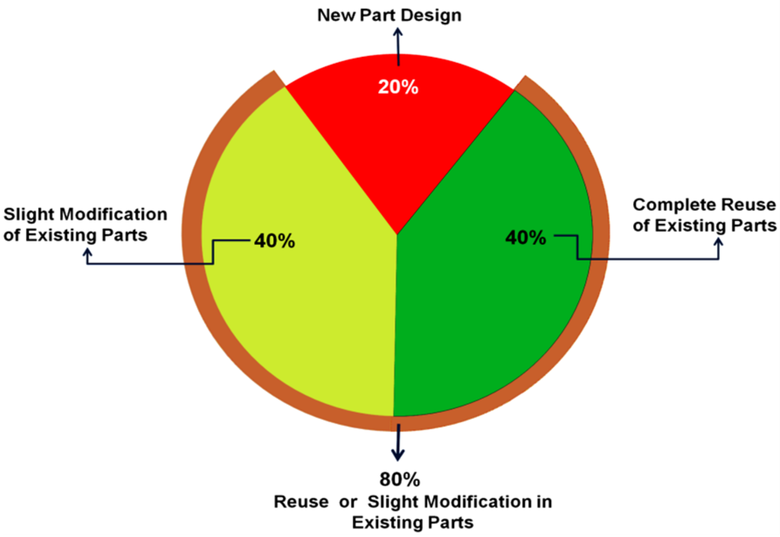


Figure 2: Distribution of design and reuse of parts in OEM (adopted from Chung et al, 2005)

Therefore, there is a need to (i) enable the reuse and exploitation of existing information and its conversion into applicable knowledge rather than redesigning and reinventing new information and to (ii) develop an infrastructure to prevent / minimise knowledge loss. The research focus here is on the knowledge enabled enterprise for rapid realisation of products, processes and automation resources in order to reduce the time and effort for design, reconfiguration and integration of automation resources.

1.3 Research Description

1.3.1. Hypothesis

An ontological knowledge based system can be used to represent the scattered knowledge of the assembly line design / reconfiguration activity in order to semi-automate some of the manual effort in the said activity in the automotive sector to improve the overall engineering process. A systematic knowledge capturing, modeling, mapping and axiomatic process can enable the optimum utilisation of available information for the efficient design and implementation of reconfigurable automation systems in automotive organisations.

1.3.2. Objective

The principal objective of the research is to design, develop and 'proof-of-concept' prototype a cross-functional knowledge based system in order to provide a decision support infrastructure and a more efficient modular approach for the rapid design / reconfiguration of assembly lines. This is to be achieved by establishing relational constraints among PPR facets so that rapid parameterised analysis can be carried out.

1.3.3. Solution

This research aims to avoid the need for the time consuming virtual verification (3D-simulation based) activity for product-resource constraints analysis, where possible, by adopting the new parameterised analysis approach of constraints and the capture of the results of the virtual verification activity in a reusable form, when it is performed.

Continuous and unpredictable engineering changes to products / processes' attributes against evolving business needs on a mixed model assembly line require explicit definition of dependency constraints among PPR entities. To help achieve this, the decision variables along the assembly line design / reconfiguration activity are quantified in terms of Key Characteristics (KCs) of products and these KCs are mapped directly to the machine capabilities in order to carry out a rapid parameterised analysis of product-resource constraints.

1.3.4. Concept

The system level conceptual model of the research is depicted in the Figure 3. An integrated cross-functional teamwork support environment applicable to all lifecycle phases of the product, process and resource is envisioned with the ability to rapidly evaluate the PPR constraints in a distributed and heterogeneous environment. The ontology is used for information modeling and relationships among PPR to explicitly define PPR properties, relations and inter-dependencies, first and then, establish inference knowledge to make rules and axioms for rapid evaluation of constraints among PPR. An important benefit of using ontologies for this purpose is that it enables knowledge-level reuse of information rather than data-level reuse as is the case in usual software engineering. In addition to this, it provides a homogeneous structure of information and in this research it provided a basis for the development of a knowledge based system.

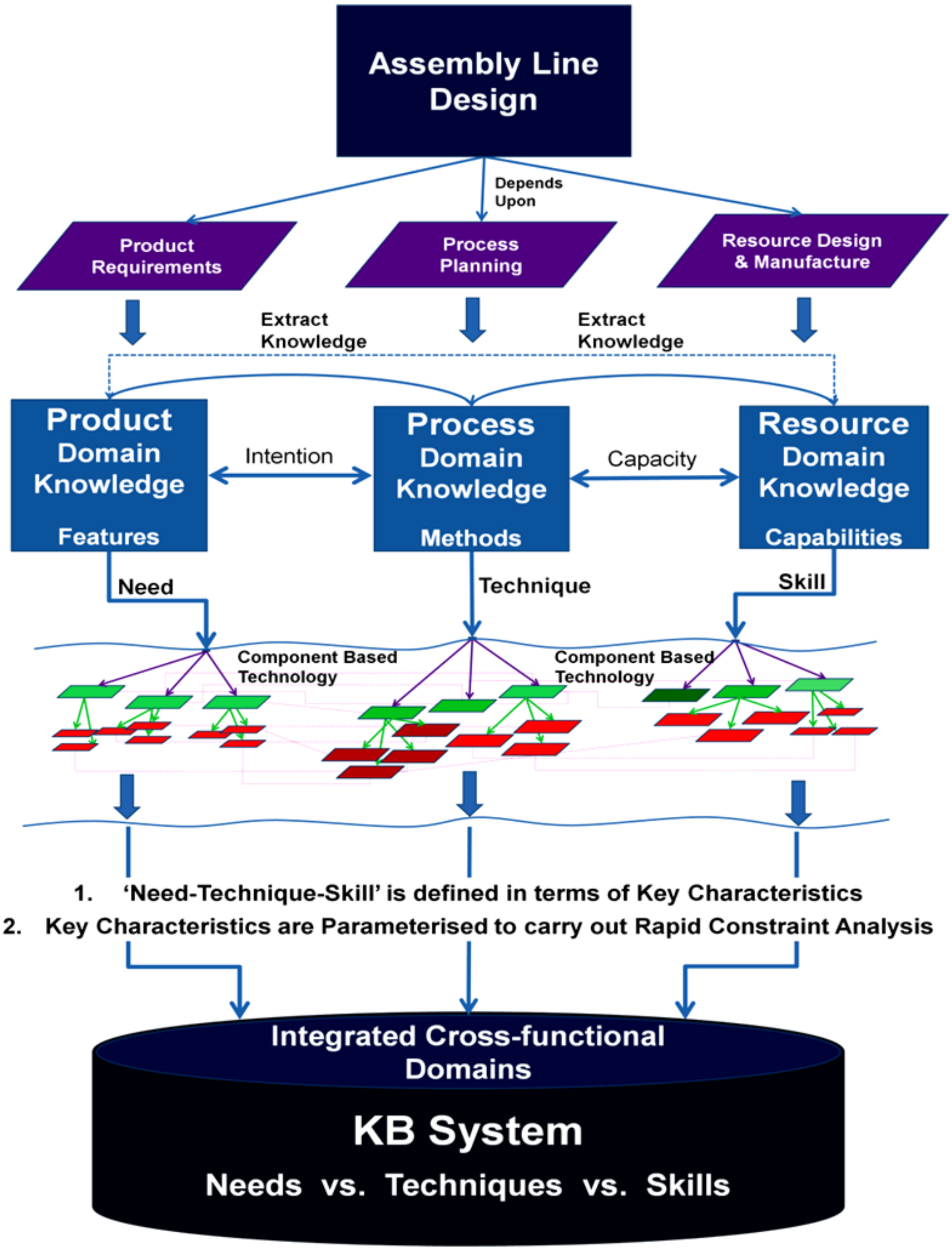

Assembly Line Design
Depends Upon
Product Requirements
Process Planning
Resource Design & Manufacture
Extract Knowledge
Extract Knowledge
Product Domain Knowledge
Features
Intention
Process Domain Knowledge
Methods
Capacity
Resource Domain Knowledge
Capabilities
Need
Technique
Skill
Component Based Technology
Component Based Technology
1. 'Need-Technique-Skill' is defined in terms of Key Characteristics
2. Key Characteristics are Parameterised to carry out Rapid Constraint Analysis
Integrated Cross-functional Domains
KB System
Needs vs. Techniques vs. Skills

Figure 3: The KB system conceptual diagram

The formulation of explicit relations and dependency constraints among product-process-resource (PPR) triples will lead to quick decision support and rapid reconfiguration of assembly machines.

1.4 Industrial Needs, Challenges and Opportunities

Product change management necessitates holistic coordination of autonomously controlled domains especially the PPR entities. The product specifications are defined in terms of its functionalities and may be disconnected to manufacturability and / or assemblability due to unavailability of explicit relational knowledge. Therefore, sometime at later stages of program management, relational dependency constraints among PPR domains are forced to be dealt with by setting the right balance between product design and machine capability, which typically results in amendments in process steps, alterations in machine design and even changes in product design specifications.

It is appreciated that assembly line design and reconfiguration is a critical and extremely complex job. The line design and reconfiguration activity is highly fragmented and distributed with geographically distant teams and decision makers resulting in lack of standardisation and consistency for the activities being carried out due to disparate nature of software tools and localised targets and responsibilities. The lifecycle dynamics of PPR domains are complex, inter-connected and inter-dependant. Knowledge is scattered in disparate domains / technology, disoriented, implicit and hard to codify. There is no standardised infrastructure, a consistent framework or a recognised support tool to address these challenges especially quantifying the implicit PPR inter-dependency constraints. The availability of explicit knowledge models of the PPR domains in the form of core competencies, capacities, limitations and their functional dependence is still a challenge within automotive organisations.

New product development or any variant of a product in production is an iterative activity of constraints evaluation and approval for production from various perspectives involving several stake holders. Currently rapid constraint evaluation is not possible due to (i) disparate applications, (ii) inconsistent data structures and (iii) lack of dependency relations and rules among PPR. It takes weeks and months to critically scrutinise constraints and thousands of man hours in a present day high-mix assembly line. This necessitates to explicitly define relational information in engineering domain especially PPR domain and represent this information in a standardised way.

1.4.1. Typical Requirements

The typical industrial requirements and challenges in the automotive powertrain sector as well as the addressed issues with the functionalities provided in the developed system are briefly discussed below:

1.4.1.1 Engine Fit Analysis - Principle Focus of the Research

Every assembly machine has limitations in its capacity to perform desired operations. Whenever there is a change in product, it requires amendments in the corresponding machine operations. It is not always possible for a changed product to be assembled on the same machine, with the same parameters / control programs or the same configuration of the machine. Similarly, there could be other changes in the product, yet the same machine with the same basic configuration may be used. The question of what changes in the product directly affect the machine and force reconfiguration or redesign cannot be readily answered by the current infrastructure within automotive OEMs. This is because analysis of this depends upon a combination of many factors including disparate information / teams, tacit knowledge, human judgement and complexity of the problem.

Ideally, the changed product must be run through the complete assembly line to check possible 'assembly hard points' (assembly hard points are the key parameters of product and/or machine which restrict smooth assembly process for a given assembly operation). This is called 'engine fit analysis', where the changed product is run through the complete assembly line. This is currently not viable through exhaustive 3D model-based simulations due to slow processing capacity of the heavy weight CAD models and associated tools, captured in proprietary formats. Therefore, it is intended to constitute a novel parameterised analysis of the PPR constraints in the form of rules and axioms to quickly evaluate the effects of product change on existing assembly machines. Thus the engine fit analysis can be transformed from an extremely time consuming, inadequately performed activity to a rapid semi-automatic knowledge based analysis for the complete assembly line. A typical use-case of converting an engine assembly line from handling a three-cylinder engine to also handling a four-cylinder engine can be quickly solved through the new knowledge based line design method.

1.4.1.2 Heterogeneous Environment

To help achieve the 'engine fit analysis' through rapid constraint-based analysis of PPR domains, information from the localised software applications needs to be exploited and shared. The PPR domains are highly distributed, consequently, the software applications are extremely diverse thus creating interoperability issues. Disparate application and distributed teams often result in conflicts in the structure and semantics of the concepts in a domain. The lack of consistent data structures is one of the main reasons for persistent disintegration of the PPR domains. It is intended to incorporate data mediation facility in the suggested KB solution so that syntactic and semantic differences may be harmonised and a global view of PPR facets be presented to the concerned teams at automotive OEMs and their supply chain partners.

1.4.1.3 Resource Capability Analysis

Recurrent changes in product design and consequent reconfigurations in automation resources and manufacturing processes necessitate optimum utilisation of available information. Once the engine fit analysis is successfully completed, the knowledge based system would provide recommendations for the required configuration of the machines. Readily accessible knowledge of former configurations of assembly line with relational constraints and dependency rules can successfully accomplish new process adjustments and predict necessary resource alterations or allocations. Products, processes and resources are related parametrically. Resources have their characteristics and capacities, therefore, a rapid and effective way to search, retrieve and select the right tool for the right task will ensure efficient design and build of the complete assembly line.

1.4.1.4 Decision Recording and Management

Once the time consuming simulation activity is completed, the lack of a proper infrastructure restricts the capture of the decisions made for the future use. Therefore, the same simulation will have to be carried out for next product change. There is a need to record decisions in the form of rules and axioms so that product-resource constraints may be calculated rapidly for the next variant of the product. The developed knowledge based system has provided such an infrastructure where the logic of the decisions made is preserved for future consultation and reuse.

1.4.2. Methodology

The research study has utilised different research techniques during different phases of the study, the major steps followed during the course of the research are summarised below:

- Problem identification and requirements analysis
- Survey of relevant tools, methods and technologies
- Design of a knowledge infrastructure
- Development of knowledge-based application tool
- Application of the solution to real world scenarios
- Refinements, improvements, implementation and evaluation

1.4.3. Empirical Study

The proof of the concept is demonstrated by a set of experiments initially carried out on a small scale test bed and later in real industrial cases. The developed system has a range of functionalities as described below:

1. Rapid assemblability analysis: requires explicit definition and evaluation of product features and machine capabilities in order to present a readymade constraint-based solution.
2. Data mediation: establishes a worldwide understandable view of concepts within a certain company.
3. Efficient selection of resources: serves for assessing the capacity of automation resources in terms of their functionalities and ensures optimum selection against desired features of the products.
4. Knowledge visualisation: provides taxonomical arrangement of the domain concepts for better understanding of the hierarchical order of complex systems.

1.5 Research Scope and Limitations

The thesis focuses particularly on the PPR facets for assembly line design and reconfiguration activity in the engineering domain of automotive powertrain organisations. Some of the other related activities and processes e.g. control programs, productivity issues and ergonomics studies are outside the scope of this work. Similarly, the research discusses the usefulness of the available knowledge, not the truthfulness of the knowledge i.e. evaluation of gathered knowledge itself. Therefore, the focus is on organisational and integrative strategies rather than the evaluation of knowledge. The machine internal behaviour is not considered i.e. how pneumatics, hydraulics or electronics work within the machine, only the function of the resources is taken into account.

1.6 Thesis Outline

The rest of the thesis is arranged as follows:

The second chapter provides a complete literature review of the subject area, similar projects and research efforts and limitations of the similar efforts. This includes topics which are directly and indirectly related to the research.

Chapter three demonstrates limitations in Ford's current business processes especially in the engineering and manufacturing domain. The current process and activities for assembly line design and the basis of introducing knowledge layer concept. Also a brief overview of the BDA project, the CCE tool with its benefits and limitations and the justification of the KB system is elaborated.

The fourth chapter describes the proposed research concept. The ontology constructed in the research is summarised with major concepts. The underlying architectural framework for PPR mapping, relational constraints and the formation of rules and axioms is explained in this chapter. This is accompanied by the ALDIMS tool description and potential impacts on the current practices.

Chapter five provides the application and implementation of the KB system with system's internal structure and functional behaviour. Similarly the KB system's architectural characteristics are highlighted, the sequence of events is elaborated through sequence diagrams and finally the characteristics of the architecture along with initial population of the system are explained.

Chapter six describes case study scenarios to demonstrate the feasibility and usefulness of the research. The chapter explains the aim and objectives of the case study, implementation of the concept on a Festo rig and also in real industrial scenarios. The main use case is engine fit analysis, this is accompanied by support activities including data mediation, library characteristics and hierarchical classification along with discussion to the industrial needs of the developed system.

Chapter seven is based on evaluation of the ALDIMS tool, discussing different evaluation facets of the new approach and potential benefits, implementation analyses and comparisons of the 'As-Is' and 'To-Be' approaches in terms of time, cost and quality as well as the viability of the new suggested approach and finally strengths, weaknesses and limitations of the developed system.

Chapter eight concludes the research study with research reflections, research contributions both on scientific and industrial levels and future work on the developed system to extend the system for wider activities within assembly line design / reconfiguration.

CHAPTER2. LITERATURE REVIEW

2.1 Introduction

This chapter explores relevant manufacturing research areas in the automotive sector as well as closely related research topics in other domains within the context of this thesis. The survey includes a range of manufacturing related topics, however, two areas best describe the bedrock of this research and are the focus of this review. These topics are (i) modular and reconfigurable automation systems (ii) ontological techniques in manufacturing knowledge based systems. These two topics have been comprehensively studied, descriptive and evaluative investigative approaches used and comparative analyses of similar research areas have been described in order to establish the foundations of the current study.

The thesis is based on the hypothesis that the current process of assembly line design and reconfiguration needs the introduction of knowledge engineering principles and techniques. This will ensure the design and development of assembly automation resources quicker, more reliable and better integrated with other parts of the enterprise. In order to address this hypothesis in the domain of reconfigurable manufacturing and assembly systems, with the impact of KB systems along with the lifecycle of PPR entities, it is necessary to revise and personify contemporary manufacturing paradigms and their evolution with a focus on key strategic issues in the automotive sector. This will eventually lead into the core research area of knowledge and ontology engineering and finally to investigate creation of a basic infrastructure for a suitable knowledge layer.

The author's research is focused on the more efficient use of automotive production line resources driven by the business need to remain competitive in the global arena. Future manufacturing scenarios have to combine the highest productivity and flexibility with minimal lifecycle cost of manufacturing equipment and organisational competences in knowledge and innovation management (Haegele et al, 2005). This is one of the key messages found across most studies of future needs (CME[3], 2004; FutMan[4], 2003; HM Treasury[5], 2004; KPMG[6], 2004; ManuFuture[7], 2006; ManVis[8], 2005; NISTEP[9], 2005).

[3] Canadian Manufacturers and Exporters

The literature survey can be summarised into three aspects:

- The general status of modular reconfiguration automation approaches in the automotive industry.
- The significance of ontological knowledge based systems in the automotive sector.
- The limitations of the existing systems.

2.2 Information Flow at Automotive OEMs

In order to design a suitable knowledge based system, it was necessary to capture and gain a detailed understanding of data producers, data consumers and format/flow of the data within Ford especially for defining and using Bill of Process (BoP) for engine assembly lines. Communication across technology intensive organisations has never been a straight forward process. This also applies to Ford as the information flow and inter-departmental communication is a complex process with several geographically distributed teams and hundreds of localised software applications being used.

Capturing information flow at Ford for powertrain line design and reconfiguration was a challenge. The information flows among both formal (Manufacturing Engineering, Machine Builders etc.) and informal (Program Approval, Supplier Selection, Control Engineering etc.) groups. There is a thin line between formal and informal groups which becomes vague or may even disappear depending upon the responsibilities allocated and goals assigned by the Simultaneous Engineering (SE) team. The complexity of the networked cross-functional team dynamics and therefore the boundary-less flow of information is a challenge to capture and model. The flow of information is goal-oriented rather than group-oriented. A condensed form of distributed nature of the information flow with prime responsibilities is shown below in Figure 4.

[4] The Future of Manufacturing in Europe: 2015-2020, http://ec.europa.eu/research/industrial_technologies/pdf/pro-futman-doc1-final-report-16-4-03.pdf
[5] HM Treasury: Science and innovation: working towards a ten-year investment framework http://www.hm-treasury.gov.uk/d/science_406.pdf
[6] Industrial and Automotive Products: Globalisation and Manufacturing http://www.kpmg.co.uk/pubs/Global_Manu_Survey.pdf
[7] ManuFuture (2006b) “Strategic Research Agenda : The future of manufacturing in Europe
[8] ManVis Report no. 6 Manufacturing Visions, http://forera.jrc.ec.europa.eu/documents/PolicyPaper_final.pdf
[9] National Institute of Science and Technology Policy http://future.wikia.com/wiki/NISTEP_report

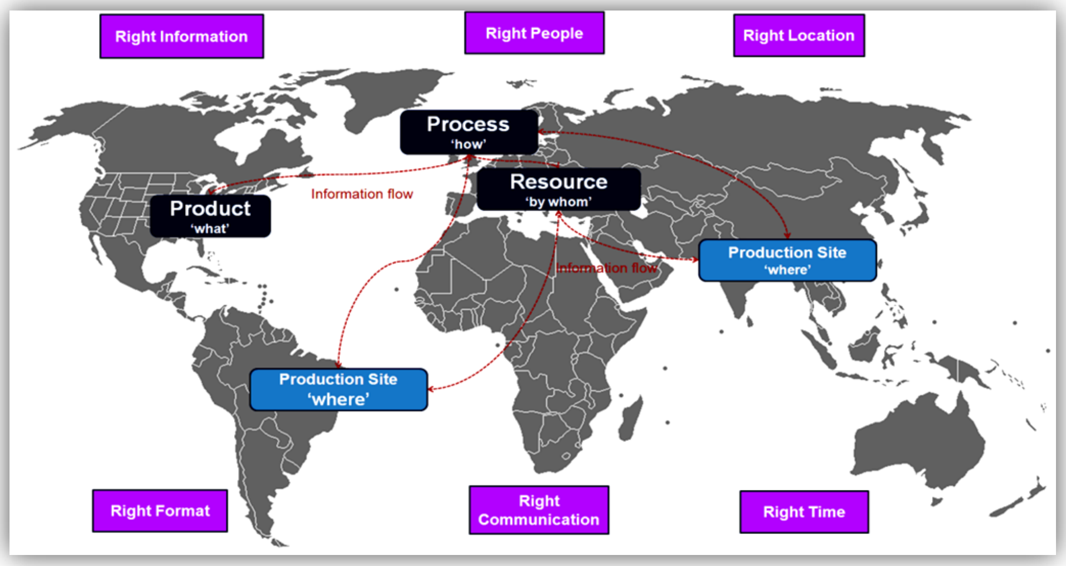


Figure 4: The PPR domains are highly distributed

Globalisation restricts information visibility, supply chain hides the knowledge and distributed sites limit the scope of collaboration. The author has had the opportunity to get first hand information from Ford engineers regarding different teams, software applications and major business activities involved in the assembly line design and reconfiguration activity.

2.2.1. Product Process and Resource Relationship

The design / reconfiguration of powertrain assembly lines depend upon three important entities i.e. products, processes and resources. The PPRs have a direct relationship with each other. Products that are manufactured cost efficiently have a resource supported process present i.e. the selection of an appropriate manufacturing process has a direct relationship with the product and resource. Products govern the final process planning and the processes shape the resources to be manufactured. In the engineering domain products, processes and resources are inter-connected as shown below in Figure 5.

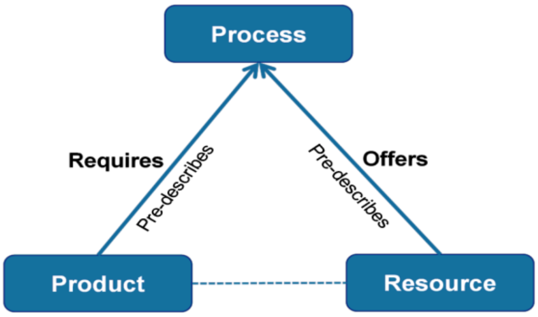


Figure 5: Processes link products with resources

Several efforts have been made to integrate PPR domains on an individual and organisational basis such as the '4P2C' prototype by Kwan and Namchul (2005). 4P2C stands for product, process, project, participant, cost, and collaboration in product development domain for Collaborative Product Development Management (CPDM) system. The prototype is meant for calculating better cost estimates. Tay and Ming (2001) proposed a similar system focussing on Virtual Design Studio (VDS) through a cyberCAD module and web accessibility. Siemens Teamcenter, IBM PLM solutions, DELMIA from Dassault Systemes and SAP[10] have similar solutions for integrating PPR domains. However, there is no mechanism to show inter-dependency or the relationships amongst PPR. This is one of the research motivations that is lacking in the prevalent state-of-the-art in automation systems research. The PPR in the engineering domain of a technology organisation are correlated, however, there is little evidence that the said entities have been relationally and axiomatically integrated especially for the assembly line design / reconfiguration activity in the real industrial scenario.

To match product specifications with the process requirements and in turn to the resource capabilities, there are a few techniques being prevalent in the manufacturing sector such as generic BoM and generic BoP concepts which are discussed briefly in the next section.

2.2.1.1 The Generic Bill of Material Concept

To improve production using the core PPR concept initiatives, a pre-defined configuration model of product, described by van Veen (1992), is generally adopted. A popular example of such a configuration model approach is the concept of the generic Bill of Material (BoM). The concept is to make a general bill of material for a product of families and all the products' bill of material should be a subset of this generic BoM. This generic BoM must contain all the bill of materials down to off-the-shelf items required to manufacture a specific assembly of product. The generic BoM concept described by van Veen (1990) aims to provide many possibilities for the modeling of large varieties of product types and their product structures without requiring large amounts of data redundancy (Claesson, 2006).

[10] www.sap.com

2.2.1.2 The Generic Bill of Process Concept

This concept is similar to generic BoM in process planning doamin i.e. to produce a standardised generic Bill of Process (BoP), within a specific domain. The BoP provides a process centric view providing many possibilities of carrying out a specific manufacturing / assembly process. BoM provides product break down structure, however, it is not sufficient especially in the assembly automation systems. Process is an important aspect, especially in assembly automation systems that link products to their manufacturing resources. Therefore BoP / generic BoP concept has been introduced with reference to the PLM system (Raza and Harrison, 2011b).

2.2.1.3 The Generic Bill of Resource Concept

The BDA project has introduced the generic Bill of Resource (BoR) concept. The project outcome has initiated a standardised library of pre-validated modules of machines which can be assembled together to design and build new / altered automation resources in automotive industry[11].

2.2.1.4 Integration of BoM, BoP and BoR

Integration of part-centric BoM, process-centric BoP, and resource-centric BoR within an environment that encourages the evaluation of alternatives and the development, storage, and comparison of temporarily infeasible scenarios is much needed in manufacturing industry (Kwan, 2005, Raza and Harrison 2012a). The greatest benefit of BoM and BoP integration would be improved production efficiencies derived from more tightly designed and managed production systems (Sly, 2004). The BoM, BoP and BoR complement each other and are required at different lifecycle stages of a product. New or modified product launch times have been reduced by 25 to 50% primarily due to the availability of instant information and to the benefits of editing existing similar processes, rather than re-engineering from scratch as is often done (Sly, 2004). The purpose of the current research is to provide an infrastructure so that BoM, BoP and BoR can be relationally integrated.

[11] http://www.lboro.ac.uk/departments/mm/research/manufacturing-systems/dsg/index.htm

2.3 Modular Manufacturing / Assembly Automation Systems

The impact of the need to support FMS can be seen in the emergence of modular approaches to manufacturing. The modular approach extends across the whole production ecosystem and can be seen as directly linked to the demands of globalised business. A substantial literature stream suggests that many products are becoming more modular over time (Fixson and Park, 2008). Reuse and modularity are important principles for improving productivity (Harrison et al, 2006). Baldwin and Clark (2000) pointed out that the issue of modularisation involves modularity in design, modularity in use, and modularity in production. Erixon (1998) defines modularisation as decomposition of a product into building blocks (modules) with specified interfaces, driven by company-specific reasons. A module is a structurally independent building block of a larger system with well-defined interfaces. A module is fairly loosely connected to the rest of the system allowing an independent development of the module as long as the interconnections at the interfaces are well thought of (Baldwin and Clark, 2000 & Ericsson and Erixon, 1999). The author has explored modularity in the current research perspective of PPR entities.

2.3.1. Product Modularisation

Modularity in products is driven by functional structure i.e. decompose the product on the base of the functions performed by the modules (Ulrich, 1995; Fixson, 2005). Modularity in products range from aircrafts (Fujita & Yoshida, 2004) to machines (Germani & Mandorli, 2004). Product modules of automobiles examples are available such as Faurecia of France builds seats, cockpit and door modules, Kuester of Germany builds door modules[12], window regulators and cables, Lames of Italy builds window regulators, Magnetti Marelli of Italy builds cockpit modules, Textron of UK builds cockpit modules[13], TRW-Lucas Varity of UK builds wiring harnesses, Visteon of UK builds cockpit, door modules and wiring etc. {source: several research articles, CIM Data, AM Data, Automotive journal articles etc.}

2.3.2. Process Modularisation

Only a small number of studies focus exclusively on the modularity of processes. It must be recognised that the problem of flexibility is a combination of product and process, and the

[12] www.autocar.co.uk

[13] www.imvpnet.org/

integration of the two will directly address many of the problems currently encountered (Marshall & Leaney, 1995). Upton and McAfee (2000) in their study on the role of information technology in manufacturing, suggest modularity as a key process feature to allow for continuous improvement. Similarly, modularity of processes is what Leger and Morel (2001) argue for more flexibility. Connecting both product and process, Watanabe and Ane (2004) find that product modularity increases the processing flexibility of machines, and in turn, the agility of a manufacturing system.

2.3.3. Resource Modularisation

The use of modular approaches to the design of automated systems allows quicker, cost-effective deployment of solutions, and future reuse of components as processes change. The European MOSYN (Modular Synthesis of Advanced Machine Tools) project (Zatarain et al, 1998), lead by the Hannover University, looked at customer-specific configurations of modular machine tools.

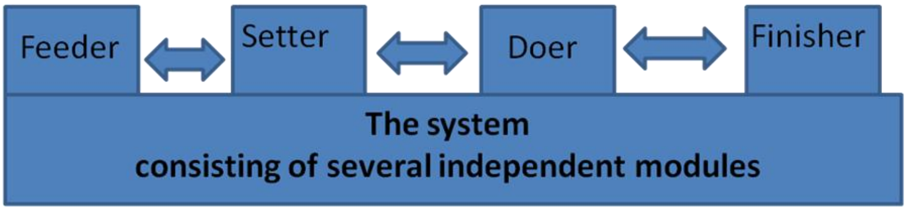


Figure 6: Hypothetical machine system example

Figure 6 shows the concept of modules for a general machine. The machine can be broken down into functionally independent modules which make it a modular machine. Any module can be replaced, if required, with the new one keeping the machine running all the time. In addition to this, the Doer module, for example, may be replaced with another function and the same machine can be used for a different operation for the same family of machines. Baldwin and Clark (2000) pointed out that the issue of modularisation involves "modularity in design," "modularity in use," and "modularity in production" (primarily focusing on "modularity in design").

2.3.4. Reconfigurable Manufacturing / Assembly Systems

Modular approaches to manufacturing can be seen as best expressed in the Reconfigurable Manufacturing System (RMS) approach. Manufacturing has evolved from dedicated to flexible and from flexible to reconfigurable manufacturing systems. The RMS is a new paradigm the aim of which is to enhance responsiveness to market changes by rapidly and cost-effectively adjusting production capacity and functionality (Koren et al. 1999; Mehrabi et al. 2000). A manufacturing system must be able to dynamically change its configurations, in terms of its own structure as well as functional principles (Mehrabi et al, 2002). The main components of RMS are Reconfigurable Machine Tools (RMTs), a new type of modular machine with a changeable structure that allows adjustment of its resources e.g. adding a second spindle unit (Koren et al, 1999). In addition to RMTs, also reconfigurable controls that can be rapidly changed and integrated in open-architecture environment are critical to the success of RMS (Mehrabi et al, 2002).

The key feature of RMS, also called changeable systems, is that unlike dedicated manufacturing systems and flexible manufacturing systems, its capacity and functionality are not fixed. The RMS will be designed through the use of reconfigurable hardware and software such that its functionality and capacity can be changed over time and unlike the other manufacturing systems, it does not have fixed hardware/software (Koren et al, 1999; Koren and Ulsoy, 2002; Landers et al, 2001; Mehrabi et al, 2002). The RMS goal is summarised by the statement, 'exactly the capacity and functionality needed, exactly when needed'. Ideal RMSs possess six core RMS characteristics: modularity, integrability, customized flexibility, scalability, convertibility, and diagnosability (Koren et al, 1999; Koren and Ulsoy, 2002; Landers et al, 2001; Mehrabi et al, 2002)

Figure 7 describes the economic goals of various manufacturing paradigms including mass, lean, flexible and reconfigurable manufacturing (Mehrabi et al. 2000).

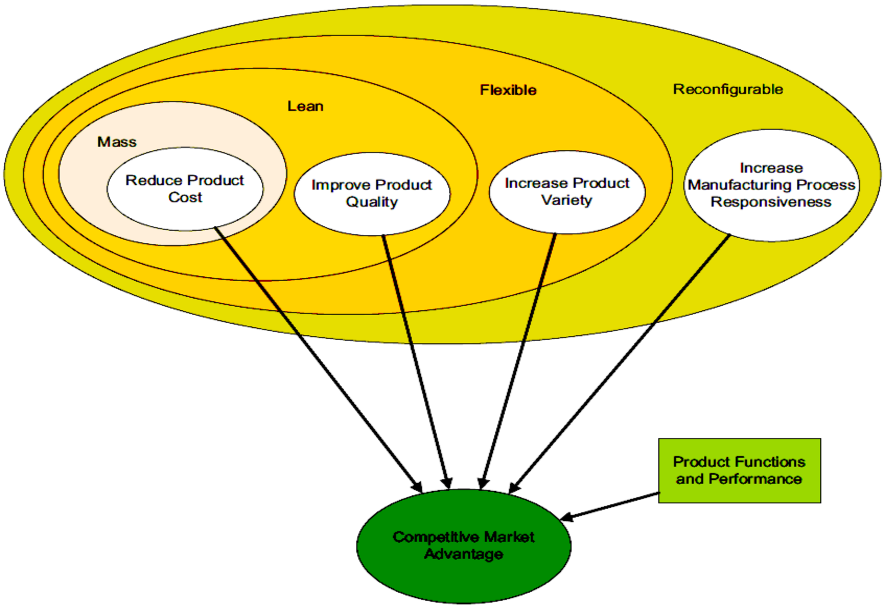

Reconfigurable
Flexible
Lean
Mass
Reduce Product Cost
Improve Product Quality
Increase Product Variety
Increase Manufacturing Process Responsiveness
Product Functions and Performance
Competitive Market Advantage

Figure 7: Manufacturing paradigms vs economic goals (Mehrabi et al. 2000)

The components of RMSs as defined by (Koren et al. 1999) included reconfigurable machines, processes, software and reconfigurable controllers, as well as methodologies for their systematic design and rapid ramp-up (Koren et al. 1999; Mehrabi et al. 2000).

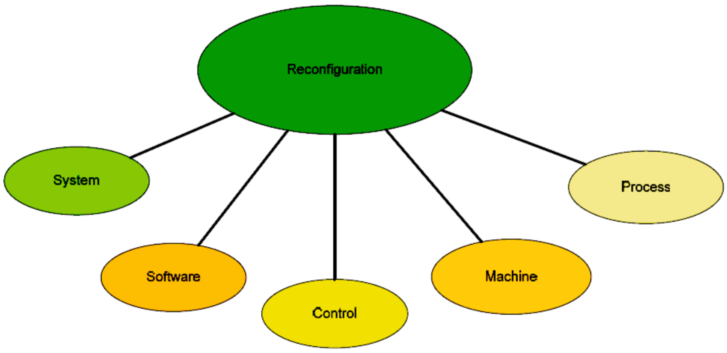


Figure 8: Reconfiguration scenarios (Mehrabi et al. 2000)

Koren (1999) describes an RMS as a system designed from the beginning for rapid changes in structure, as well as in hardware and software components, as depicted in Figure 8, in order to quickly adjust production capacity and functionality within a part family in response to sudden changes in market or in regulatory requirements.

In FMSs general flexibility is provided by the use of equipment with built-in high functionality, RMSs customised flexibility is provided by scalability and reconfiguration as needed and when needed (Mehrabi et al., 2000). Flexibility is then a key factor also in RMSs. RMS is a manufacturing system with customized flexibility and FMS is a manufacturing system with general flexibility (Hu, 2005). In this sense, many authors notice that there are common grounds in philosophy between the FMS and RMS paradigms and support the idea that they represent a continuum, so that they predict that the future RMSs will be more flexible (Stecke, 2005). RMS amalgamates best characteristics of Dedicated Manufacturing Lines (DML) and Flexible Manufacturing Systems (FMS). A comparative analysis of dedicated, flexible and reconfigurable systems is summarised in Table 2.

Characteristics	**Dedicated**	**Reconfigurable**	**Flexible**
Machine Structure	Fixed	Adjustable	Fixed
System Focus	Part	Part Family	Machine
Scalability	No	Yes	Yes
Flexibility	No	Customised	General
Simultaneous Operation Tool	Yes	Yes	No
Productivity	High	High	Low
Lifetime Cost	Low	Medium	High

Table 2: RMS combines features of dedicated and flexible systems (Koren et al, 1999; Dashchenko, 2003)

The recent Delphi study, Visionary Manufacturing Challenges for 2020, conducted by the USA's National Research Council has identified reconfigurable manufacturing as first priority among "Six Grand Challenges" for the future of manufacturing (Bollinger et al., 1998). Various aspects of RMS are now under investigation by researchers around the world, one of the most important one is developing a framework for constructing reconfigurable systems and one of these frameworks is knowledge based reconfiguration techniques in the manufacturing enterprises, the current research is addressing this area.

2.4 Knowledge Based Manufacturing Strategies

The manufacturing process is heavily dependent on the knowledge of individual engineers and production operatives. This reliance extends into all the areas of manufacturing from product design, process planning, machine design and build to line design / reconfiguration. In the wider enterprise, systems have evolved to largely automate related activities such as logistics planning and resource management. Within recent years, this has extended into the production line and is an emerging area of future research work. A few most relevant manufacturing strategies related to the current research are briefly discussed in the next sections.

As a direct attempt to bring in automated knowledge into manufacturing, the Computer Integrated Manufacturing (CIM) management philosophy was developed. Here the functions of design and manufacturing are rationalised and coordinated using computer, communication, and information technologies (Bedworth et al., 1991). Simply, CIM is the use of computer systems to integrate a manufacturing enterprise (Aletan, 1991). The current research falls under the CIM category and it directly addresses the integration issues of the CIM philosophy through ontological representation and structuring of domain entities and concepts.

The aim of the set of software tools and methodologies for Digital Factory concept is to comprehensively design, model, simulate, evaluate and optimise products, processes and systems before a new factory is built or any modification is actually carried out on an existing system, in order to improve quality and reduce time of planning processes (Bracht et al, 2005; Kühn, 2006; Chryssolouris et al, 2009). The digital manufacturing concept from modeling and integration of knowledge is an important part of the current research.

2.4.1. Reusable Manufacturing Knowledge

Building on the work around CIM and digital factories, the concepts of reusable manufacturing knowledge have linked the use of digital technologies to the capturing of knowledge from executed production processes. Recently, knowledge has been proclaimed a strategic tool in automation by many scientific communities and researchers (Bontis and Choo, 2002; Mcevily, and Chakravarthy, 2002; Alizon and Shooter, 2005; Raza et al, 2009b; Raza and Harrison, 2011f). The wider world is increasingly moves towards the knowledge based economy predicted by Bell (1974), this has led to a technological expansion and different techniques and methodologies have been introduced, devised and implemented in different manufacturing and assembly scenarios.

Manufacturing is one of the dominant sectors of EU economy. The EuroKnowledge[14] is an initiative to encourage, co-ordinate, disseminate, promote and undertake activities within Europe related to the standardisation of knowledge based systems and knowledge reuse and knowledge level representations, domain ontology expression and criteria[15]. Many similar initiatives have been undertaken related to EuroKnowledge. For example ManuFuture, future of manufacturing in Europe: a vision for 2020, (FP6 framework) proposes High Value Manufacturing (HVM). ManuFuture emphasises an integrated knowledge community in manufacturing. The FP7 framework proposition is building the Europe of Knowledge, transition to knowledge-based industries and factories as well as integration of technologies for industrial applications[16]. It suggests development of new industrial models and strategies taking it further from the Lean. Manufacturing in Europe currently has to be knowledge-intensive given the European demands for high-tech products (Wijnhoven, 2005; Bernard & Tichkiewitch, 2008; Flegel, 2010). The ManuFuture value proposition consists of knowledge based and driven, high value added, life cycle oriented technologies beyond borders. The said proposition suggests knowledge based manufacturing for competition in the present world, top priorities set by ManuFuture (2006) are:

- Knowledge based and driven
- New industrial models and strategies
- Adaptive production systems

[14] http://www.aiai.ed.ac.uk/project/euroknow/

[15] http://cordis.europa.eu/esprit/src/9806.htm

[16] 7th FP: Building the Europe of Knowledge/NMP
http://www.rm-platform.com/index2.php?option=com_docman&task=doc_view&gid=453&Itemid=1

- Networked and High Value Added (HVA) production
- Convergence of technologies
- Information and communication technology

A similar research based in Germany is KAP (Knowledge, Awareness and Prediction of Man, Machine, Material and Method in Manufacturing) which has accentuated the need for a knowledge based manufacturing for ultimate success[17]. Claesson (2006) has described the project as sharing knowledge among different stake holders to help not only enterprise integration but also as an enabling technology for reconfiguration and reuse of parts and resources to reduce investment costs. Later it describes adaptive manufacturing, ICTs and especially the importance and far reaching effects of knowledge based engineering.

2.5 Knowledge Engineering

In the past decade, there has been a widespread interest in the field of Knowledge Engineering (KE) techniques, which are able to simulate human expertise in narrowly defined domain during the problem solving by integrating descriptive knowledge, procedural knowledge and reasoning knowledge (Hendriks, 2001; Huber, 2001; Liebowitz, 2001; Tiwana, 2001). Knowledge has been defined as information combined with experience, context, interpretation and reflection (Davenport et al., 1998). KE is the study that constructs computable models in relation to ontology (representational mechanism) and logic (inference mechanism) to solve some practical problems in the different application domains (Sowa, 2000). Knowledge engineering is, therefore, defined as the branch of engineering that obtains knowledge about some knowledge intensive subject and transforms it to computable form for some useful purpose (Sowa, 2000).

The paradigm beyond ‘lean’ will be driven by the availability and exploitation of real-time knowledge across the manufacturing enterprise to optimise the value chain from the suppliers through manufacturing plants, and into the distribution channel (Slotwinski and Tilove, 2007). Knowledge may for instance include person-dependent skills, explicitly described insights (like explanations, formulas, designs, predictions, and patents), effective work procedures, rules and methodologies, and databases (Jetter, 2005). Knowledge is a unique competitive force, it is a core competence and provides an organization with sustainable competitive advantage (Quinn, 1992; Davenport & Prusak, 1997; Remus, 2003; Maier, 2004). The central theme in knowledge

[17] http://cordis.europa.eu/fetch?CALLER=PROJ_ICT&ACTION=D&CAT=PROJ&RCN=95347

engineering techniques is the conceptual modeling of the system in the analysis and design stages of the development process.

Looking towards 2020, the HVA paradigm states the need to integrate R&D knowledge into continuous generation of HVA products and processes. Recently completed Integrated Manufacturing Technology Roadmapping (IMTR) initiative launched in the United States, highlighted six key themes, useful for businesses within manufacturing sectors highlighting knowledge management as the most important one, showing how technology developments are likely to converge towards the 'information driven seamless enterprise' i.e. aligning knowledge assets and knowledge management initiatives with business objectives (IMTR, 1999).

2.5.1. Knowledge Based Systems

A Knowledge Based System (KBS) is defined as "A computer system that is programmed to imitate human problem-solving by means of artificial intelligence and reference to a database of knowledge on a particular subject" (Computer User High-Tech Dictionary, 2004). A knowledge-based system is a system that can undertake intelligent tasks in a specific domain that is normally performed by highly skilled people (Miresco and Pomerol, 1995). A KBS is developed using knowledge engineering techniques (Studer et al, 1998), these are similar to software engineering techniques, but the emphasis is on knowledge rather than on data or information processing. Many of the knowledge engineering methodologies developed emphasise the use of models e.g. Common KADS, MIKE, Protégé etc. (Gabriela, 2005).

In the early stages, knowledge-based systems were built using the knowledge of one or more experts, essentially, a process of knowledge transfer (Studer et al, 1998). Nowadays, a KBS involves "methods and techniques for knowledge acquisition, modeling, representation and use of knowledge" (Schreiber et al, 1999). Ontologies and problem-solving methods enable the construction of KBSs from components reusable across domains and tasks. Decisions are being made under pressure and cost and time invariably dominate the decision making process (O'Brien, 1998). Successful KBSs can improve the efficiency and quality of information systems by an order of magnitude and their advantages are well documented (Hayes-Roth and Jacobstein 1994). Decision Support Systems (DSSs) can assist managers in making strategic decisions by presenting information and interpretations of various alternatives (Turban et al, 2007). Pal & Palmer (2000) have classified DSS as (i) rule based, (ii) case base and (iii) hybrid systems. The DSS help engineers in decision making process, the engineer may remediate after consulting the DSS (De Vin et al, 2006).

A few of the major differences between a knowledge based system and a conventional system are shown in tabular form below in Table 3.

KBS	Conventional Systems
Domain modeling and problem solving approach	Information management approach
Separated knowledge from control	Amalgamation of knowledge and control
Ability to reason heuristically	Algorithm processing
Symbolic	Numeric processing

Table 3: Comparison between KBS and Conventional Systems

2.5.2. Main Components of a Knowledge Based System

Brinkop et al. (1994) has defined knowledge based system as an equation i.e.:

Knowledge Based System = Domain-Independent Inference Engine + Domain-Specific Knowledge Base + Problem-Specific Database.

The knowledge based system keeps factual knowledge of the domain separate from the problem solving knowledge. The factual knowledge consists of concepts and their relationships while the problem solving knowledge consists of specific conditional facts e.g. axioms and rules etc. The knowledge based systems provide a means to answer the ‘what-if’ and ‘why-not’ questions and scenarios. In general, most KBS can be considered to consist of four main components as described by Oliver (1994).

2.5.2.1 Knowledge Base / Knowledge Database

The component of a KBS that contains all the information associated with the domain to which the system is applied. This information may be documented definitions, facts, rules and heuristics. Knowledge bases may be organised hierarchically as knowledge trees or as sets of rules. The knowledge should be able to be viewed and manipulated independently.

2.5.2.2 Context / Fact Base

Its content changes dynamically and includes information that defines the parameters of the specific problem and information derived by the system at any stage of the solution process. This is the component of a KBS that contains all the information about the problem currently being solved.

2.5.2.3 Inference Mechanism

The inference mechanism uses the knowledge base to modify and expand the context in order to solve a specific problem. This is the component of a KBS that controls the reasoning process of the system.

2.5.2.4 User Interface

In addition, a user interface is essential in allowing users to operate the system in a simple and easily followed manner using whatever control items and methodologies are required. In commercial systems, it is not unusual for the development of the user interface to take up to 70% of the total development effort (Sutcliffe, 1988). A knowledge acquisition module may be considered to be a part of the user interface, allowing the users and/or system developers to enhance the scope and breadth of the knowledge bases within the system.

2.5.3. Knowledge Representation in KBSs

Miles and Moore (1994) have described methods of knowledge representation in knowledge based systems as below:

2.5.3.1 Rule based

Rule-based representation schemes utilise a set of rules to store the domain knowledge, sometimes known as production rules. Rule based method of representing knowledge is also called logics.

2.5.3.2 Frame based

These systems employ a representation of the knowledge of the problem concerned, either utilising slots on objects/frames, or nodes and their interconnections in a network.

2.5.3.3 Logic Based / Predictive Calculus

In logic-based systems knowledge is represented as assertions in logic. Logic based languages allow quantified statements and other well defined formulas as assertions. The flexibility of mathematical logic makes these knowledge representation systems powerful (Marcellus 1989; Konigsberger and De Bruyn, 1990; Moula, 1993), however the difficulty in handling uncertainty make them unsuitable for some applications.

2.5.4. Selection of Tools for Building a KBS

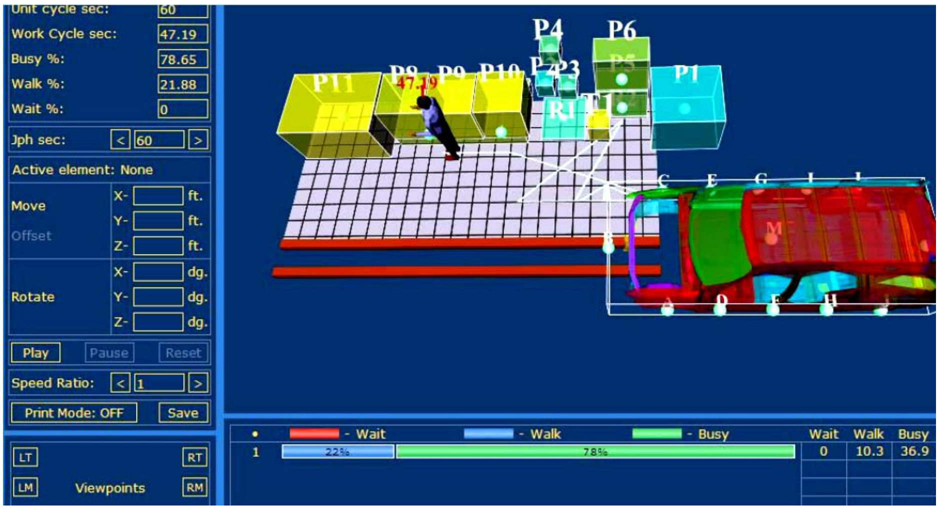
A KB system can be built by a wide range of methods, these include:

- Expert system shells, JESS and CLIPS
- AI programming languages like LISP, PROLOG and OPS5
- Conventional languages such as C, VB/VC, PASCAL, etc.
- Ontology languages such as OWL and WSML
- Standardised Template Driven e.g. MOKA and KOMPRESSA.

Facts, rules, heuristics and inference strategy are used in almost all types of KB systems, however, ontologies are one of the best methods to develop a KBS due to the built-in capability of reasoning and inferences to derive new knowledge from already available facts, other than the fundamental requirements to build a successful KBS. Therefore, ontologies have been used to build the current KB system. KOMPRESSA employs IDEF0 technique while MOKA uses templates to capture and organise knowledge. However, it is not feasible for complex problem solving activities because the rigid nature of templates restrict the tool to be used for complex knowledge modeling situations as dealt in this thesis. Also the integration to the source database is not possible, though, there are fields to declare where that knowledge source came from yet it cannot be linked

2.5.5. Knowledge Based Systems in Manufacturing / Assembly

There are many similar research efforts undertaken in recent past as well as ongoing research projects currently under progress have been surveyed and a brief description is provided in this section. Quite a few tools, popularly called KBS / DSS, have been developed in this regard (Eddy et al, 2000; Faura et al, 2001; Grabot et al, 1996). One of these tools is the e-Workcell which is used at Ford for optimised workstation layout to reduce non-value added time by finding mathematically optimal solution to efficiently layout the workstation.



Unit cycle sec: 60
Work Cycle sec: 47.19
Busy %: 78.65
Walk %: 21.88
Wait %: 0
Jph sec: 60
Active element: None
Speed Ratio: 1
Print Mode: OFF

●	Wait	Walk	Busy	Wait	Walk	Busy
1		22%	78%	0	10.3	36.9

Figure 9: The e-workcell being used at Ford

A snapshot of e-workcell is shown in Figure 9. The e-Workcell combines the global assembly information along with specific knowledge of the local assembly plant layout to optimize the operator walk patterns in any given workstation (Klampfl et al, 2004). Toyota's intelligent planning and line allocation systems based on the Object-Oriented Expert System G2[18] by Gensym Corporation[19] as described by Ishi (2004).

The analysis of the process sheets is performed by a knowledge-based system called the Direct Labour Management System (DLMS). DLMS provides the framework for allocating the required work among various operators at the plant and builds a foundation for automated machine translation of the process descriptions into foreign languages (Klampfl et al, 2006; Rychtyckyj, 2007).

There have been several independent research efforts and a brief of the most relevant efforts have been presented here. A similar KB system developed by Lai (1993) is called Knowledge Based Design for Assembly (KBDA), written in PROLOG, performs analysis for system products and provides advice for design refinement. The user provides input parameters and an appropriate assembly method is advised for the assembly of product, the resource limitations are not taken into consideration. The computer aided Assembly Line Balance System (ALBS) adopts a

[18] G2 is Gensym's flagship product for expert system applications that can emulate the reasoning of human experts as they assess, diagnose, and respond to unusual operating situations or as they seek to optimize operations.
[19] http://www.gensym.com

procedure that an industrial engineer may follow when developing a manual line balance (Bhattacharyya et al, 1992).

Molloy (1993) presented an architecture for an integrated Computer-Aided Assembly Process Planning (CAAPP) system based on feature information. It consists of a product model (which accesses an assembly data base and a component data base), a process knowledge base, a DFA knowledge base, a feature-based CAD system, a CAAPP system, component data, process data, DFA software, and DFA knowledge acquisition software. The KCAPPS, a Knowledge-based Computer-Aided Process Planning System (Wie et al, 1990), is an integrated system for design and manufacturing planning. The four major elements of the KCAPPS architecture are the integrated data base, the user interface, the knowledge base module, and the main module, the main module provides the mechanism to infer the production rules stored in each knowledge base.

Yongqian (2007) reported manufacturing process knowledge model. Based on geometric models, relationships, and process information, the four key elements, namely parts, resources, operations, and manufacturing features are defined with four classes. The focus is on car body production line. The product is decomposed to part level, process planning is decomposed to steps, however, machine are and cannot be decomposed in the system. This system is based on PDM theory (Yongqian, 2007) and it is hard coded as this system is programmed with VC8.0 based on .NET. Wang et al (2009) has demonstrated the construction of knowledge based engineering platform for armoured vehicle. The system is the application of knowledge based engineering approach to the rapid design and analysis of an automotive structure focusing on advancements for Body-In-White (BIW[20]) engineers. A tool developed DART, Design Analysis Response Tool, to aid in analysis of BIW stage vehicle.

Milani and Hamedi (2008) reported a knowledge-based system for selecting fastening tools in automobile assembly lines. The system is a rule based expert system and is generated using a commercial expert shell called Exsys[21] written in C programming language. Sandberg and Larsson (2006) reported amalgamating KBS and CBR and presented an approach with industrial implementation potential regarding automating redesign of sheet-metal components in early product development to avoid manufacturing problems due to design flaws and non-optimal

[20] Full sized mock-up of the final design of the vehicle

[21] Exsys Corvid™ www.exsys.com/

designs. Zhang et al (2009) designed a decision support system for automotive body assembly design in conceptual design stage based on ontology and implemented in Java. Wang et al (2010) proposed a KBS for rapid product design through Knowledge Fusion of Nx by adding intelligence in the design process.

Hsiao and Huang (2002) developed a knowledge-based system to generate product-shape design based on a back-propagation neural network that builds the relationships of product-shape parameters and customer needs, and a product database that consists of design elements, product image and shape generation rules. Kiritsis (1995) proposed a knowledge based system in manufacturing environment especially for process planning. Sudin et al (2007) has described a prototype KBS for material selection in bumper beam design. ICAD system of Delft University of Technology, is a knowledge-based expert system, used to fully parametrically describe the airplane geometry (La Rocca, 2002; Nawijn, 2006). Chudoba and Huang (2006) developed a system consisting of a database system, an information-base system and a method library for aerospace conceptual design which is essentially a knowledge based system.

A knowledge based reasoning system developed by Shehab and Abdalla (2006) select the most economic assembly technique for the product ‘at an early design stage’ as well as estimates the assembly time and cost for manual, automatic, and robotic assembly methods. Lohse (2006) developed a KB system for modular assembly systems focussing on developing a holistic design theory in the form of ontologies. The framework is developed in Protégé and is named ONTOMAS (ONTOlogy for design of Modular Assembly Systems). Lanz (2010) developed a KB system based on ontologies highlighting on product and product features and associate processes that occur on the feature level.

The KB systems discussed in this section provide an outlook of the research trends in the KB system development in manufacturing / assembly automation systems. The use of ontologies for KB system development is being used over the past few years as it provides additional reasoning capabilities to the available knowledge in the domain. In the current system, both implicit and explicit knowledge is captured, whereupon it is used for reasoning on a complex task in a high-tech environment.

VEMPRO[22] is funded by the German Federal Ministry of Education and Research (BMBF) and supervised by Project Management Agency (PTKA) is an integrated development environment that allows considering knowledge about the reliability of a product in an early stage of the development cycle of multi-functional products. A holistic concept is developed that allows the virtualisation of the product concept as well as the knowledge and rule based analysis of the reliability of the developed concept.

FRAME[23] (Fast Ramp-Up and Adaptive Manufacturing Environment), funded by the EU 7th Framework Programme, aims to create a new solution for highly adaptive, self-aware assembly systems, which aims to use automated self-learning, dynamic knowledge sharing, highly integrated sensor networks and innovative human-machine interaction mechanisms. These next generation assembly systems equipped with FRAME technology will be able to proactively support ramp-up, error recovery and operational performance improvement. This will lead to a dramatic cost and time reduction of deploying and maintaining complex assembly systems on demand and improve their effectiveness.

i-CONIK[24] (An Internet-Based Collaborative Platform for Managing Manufacturing Knowledge, 2010-11) is a large-scale integrating project partially funded by the European Union's 7th Framework Programme under the Information and Communication Technologies Call “Factories of the Future” (FoF). Two main objectives of i-CONIK are (i) systematic analysis of shop floor data for process and product design specifications, (ii) automatic extraction and representation of knowledge from history of design changes, focusing on the process design and operation sequences.

The Manufacturing Engineering Laboratory (MEL) of the National Institute of Standards and Technology (NIST) has been working on the concept, design and implementation of Smart Assembly Systems. The vision of the SAS is empowered, knowledgeable people, a multi-disciplined, highly skilled workforce is empowered to make the best overall decisions.

Enabling technologies for SASs are:

1) Flexible assembly processes, equipment and tools

[22] http://www.produktionsforschung.de/verbundprojekte/vp/index.htm?TF_ID=49&VP_ID=2642
[23] www.frame-eu.org
[24] http://noam.mech.upatras.gr/i-conik

2) Accurate, easy-to-use, pervasive, persistent, virtual capability
3) Effective plant floor decision support tools, actionable information for man and machine i.e. interoperability from "shop floor" to "top floor" (Slotwinski and Tilove, 2007).

Similar efforts have been under development and implementation at Ford's Dunton Technical Centre (DTC), Ford Motor Company, UK in collaboration with Loughborough University, UK, e.g. the BDA project explained in section 3.4.1.

2.6 Ontology Engineering

Ontology is defined as "a formal, explicit specification of a shared conceptualisation (Gruber, 1993)." Formal means that the representation model is machine readable, explicit means it is unambiguously defined and conceptualisation means a model of the real world and shared means the knowledge defined and represented is accepted by a community of practice / group.

It is recognised that the ontology is an appropriate methodology to accomplish a common consensus of communication, as well as to support a diversity of activities of KM, such as knowledge repository, retrieval, sharing, and dissemination (Neches et al, 1991; Gruber, 1995). In particular, it allows communication and reuse of knowledge among different entities to share the same domain area (Pundt and Bishr, 2002). Ontology efficiently handles two separate types of knowledge i.e. factual knowledge of domain concepts (what do we know and can use to express our knowledge) and problem-solving knowledge (how do we express our knowledge).

There have been an increasing number of research projects applying ontological techniques in the context of product development (Roche, 2000; Duineveld et al, 2000; Ciocoiu et al, 2001; Lin and Harding, 2003). Ontology can be used for expressing some fundamental concepts like things, relations and events precise in some other way to manufacture an approved vocabulary for distributing data, over the World Wide Web (Noy and Klein, 2004). Providing an ordinary acceptance, Valarakos et al. (2004) said that ontology can be used to make possible distribution and reuse the information and data. The fundamental approach used to make ontology is the process specification language and web based technologies (Schlenoff et al, 2000). Usually the standards of web based technologies that contributes to manufacturing ontology are the XML, the web ontology, the RDF, WSML and XML Metadata Interchange (XMI) format (OMG, 2005). To retrieve the information from the ontology, the ontology query languages will also be needed. The basic theory of such query languages can be divided into two mechanisms: RDF-based query e.g. SPARQL and Logic/Rule-based query e.g. SWRL (Zhou, 2011).

Most ontologies have two stages of development lifecycle, (i) informal stage: ontology is sketched out using either natural language descriptions or some diagram technique, (ii) formal stage: ontology is encoded in a formal knowledge representation language, that is machine computable. Major types of ontologies include (i) generic ontologies i.e. concepts are valid across different domains, (ii) domain ontologies i.e. conceptualisation of a particular problem domain, (iii) application ontologies i.e. particular task oriented extensions to domain ontologies, (iv) representation ontologies i.e. framework to conceptualise the world. The three main capabilities provided by ontology are formal terminology to facilitate communication , semantics to assist in interoperabilityto and inference capacity to achieve reusability as shown in Figure 10.

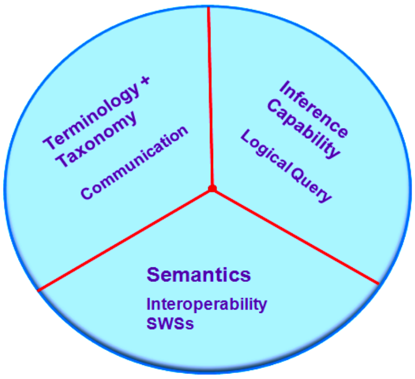


Figure 10: Three facets of ontology adopted from Uschold and Gruninger (1996)

Ontology benefits include (i) common terminology i.e. agreed vocabulary in the domain, (ii) hierarchy i.e. the concepts are arranged on different hierarchic levels, (iii) encapsulation i.e. information hiding mechanism, (iv) aggregation i.e. whole/part relationship, (v) inheritance i.e. parent/child relationship (vi) Reusability i.e. concepts are reusable (vii) scalability i.e. extendable as and when required, (viii) Reasoning and Inference i.e. derivation of new knowledge from known (ix) Axioms and Rules i.e. refinement of concepts with restrictions and constraints.

2.6.1. Ontology Languages

Ontology languages are formal languages used to construct ontologies. Ontology languages are classified on different criteria e.g. First order logic based or traditional (Common Logic, F-Logic, and KIF), mark-up based (e.g. OWL, RDF and RDFS), Description Logic based (e.g. KL One, RACER and OWL) and a combination of first order and logic programming based (e.g. WSML).

A detail of ontology languages is available on W3C[25], OMG and several other websites[26]. Resource Description Framework (RDF) developed a basic ontology language for describing web resources which are normally identified by a Uniform Resource Identifier (URI). It is a data model represented in XML syntax with simple semantics, containing objects and their relations. The RDF statements are written as a tri-tuple <Subject, Predicate, Object>. RDF Schema (RDFS) is developed as an extension of RDF. RDFS allows for defining instances of classes, subclasses of classes and sub-properties of properties. However, RDF(S) does not provide transitive, inverse or symmetrical properties (Saha, 2007).

2.6.1.1 OWL

Therefore, OWL[27] Web Ontology Language was developed by the W3C Web Ontology (WebOnt[28]) as an extension of RDF(S) with greater expressivity and automated reasoning support (Saha, 2007). XML is the foundation of this language stack because of the ability to define customised tagging schemes. RDF is located in the middle of the stack as a flexible methodology for data representation. OWL is on the top of the stack and it provides a way to formally define the terminology used in web (Sowa, 2000; Zhou, 2011). OWL facilitates greater machine interpretability of Web content than that supported by XML, RDF, and RDF Schema (RDF-S) by providing additional vocabulary along with a formal semantics. OWL includes three sub languages called:

OWL-Lite: Roughly consists of RDFS plus equality and 0/1-cardinality. It represents a migration path from other taxonomies. It is intended for classification hierarchies and simple constraints. It should be kept as simple as possible in order to facilitate the tool development.

OWL-DL: Contains the language constructs but with hierarchy restrictions. It provides computational completeness and decidability, and has a maximum expressive power within DL Description Logics fragment.

OWL-Full: Composed by the complete vocabulary interpreted more broadly than in OWL DL. The language incorporates maximum expressive power and syntactic freedom, and offers no computational guarantees.

[25] http://www.w3.org/standards/semanticweb/ontology
[26] http://en.wikipedia.org/wiki/Ontology_language
[27] http://www.w3.org/TR/owl-features/
[28] http://www.w3.org/2001/sw/WebOnt/

2.6.1.2 WSML

The WSML[29] (Web Service Modeling Language) is a relatively new ontology language with the goal to provide one coherent framework which brings together web technologies with different well known logical language paradigms. The Description Logics (Baader, 2003), Logic Programming (Lloyd, 1987), and F-Logic (Kifer, 1995) are the starting points for the development of a number of WSML language variants. The core language is based on the intersection of Description Logics and Logic Programming (Grosof et al, 2003). Syntax-wise[30], WSML takes the user point of view with, on the one hand, its syntax for conceptual modeling and, on the other hand, allows full flexibility to specify arbitrary logical axioms and constraints using the logical expression syntax (de Bruijn et al, 2005).

WSML-Core: corresponds with the intersection of Description Logic and Horn Logic (Grosof et al., 2003) (without function symbols and without equality), extended with data type support in order to be useful in practical applications. WSML-Core is fully compliant with a subset of OWL. The WSML-Core is based on the well-known DHL (Description Horn Logic) fragment which is that subset of the Description Logic which falls inside the Horn logic fragment of First-Order Logic without equality and without existential quantification (de Bruijn et al, 2004).

WSML-DL: extends WSML-Core to an expressive Description Logic, namely, SHIQ, thereby covering that part of OWL which is efficiently implementable.

WSML-Flight: extends WSML-Core in the direction of Logic Programming. WSML-Flight has a rich set of modeling primitives for different aspects of attributes, such as value and integrity constraints. Furthermore, WSML-Flight incorporates a rule language, while still allowing efficient decidable reasoning. More precisely, WSML-Flight allows any Datalog rule, extended with inequality and (locally) stratified negation.

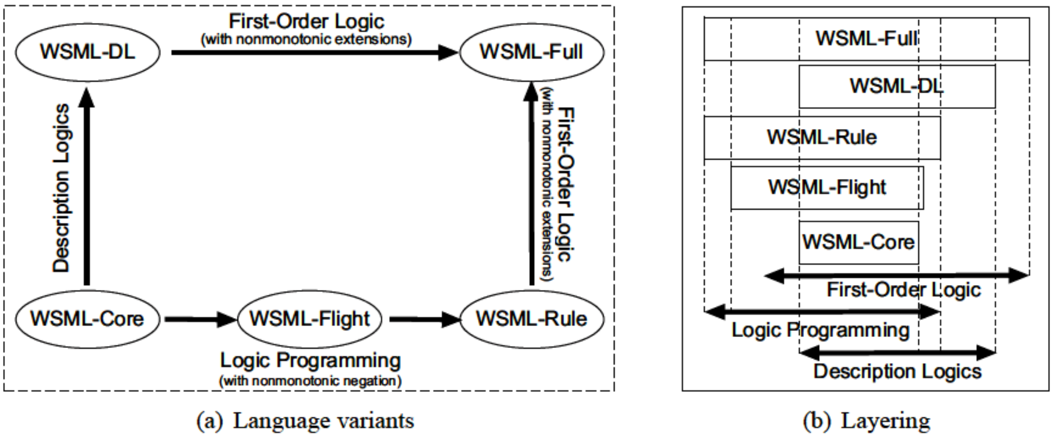
WSML-Rule: extends WSML-Flight to a fully-fledged Logic Programming language, by allowing function symbols and unsafe rules.

WSML-Full: unifies all WSML variants under a common First-Order umbrella with non-monotonic extensions which allow to capture non-monotonic negation of WSML-Rule.

[29] http://www.wsmo.org/wsml/

[30] http://www.wsmo.org/wsml/wsml-syntax

Figure 11 shows diverse variants of WSML language and relationships between them.



(a) Language variants

(b) Layering

Figure 11: WSML Variants and Layering (adopted from WSML Rule Reasoner, 2009)

The initial ontology was built in WSML DL, however, later switched to WSML Flight. WSML Flight provides one of the most robust rule languages available (Franconi and Tessaris, 2004; Lausen et al, 2005; Raza and Harrison, 2012a). WSML rule is an extension of WSML Flight by allowing unsafe rules[31] also called open world assumptions. A statement cannot be assumed true just because its negation cannot be proven (Horrocks et al, 2004). To use safe rules, WSML-Flight was chosen for the current research. WSML-Flight and WSML-Rule are rule languages based on the Datalog and Logic Programming fragments of F-Logic, respectively, extended with inequality and default negation in rule bodies under the Perfect Model semantics (de Bruijn et al., 2005).

2.6.1.3 Ontology Editors

There are several ontology editors, both commercially off-the-shelf, commercially custom-built as well as free editors for research and development purposes. A brief outlook of the editors is presented here. An ontology editor called Apollo from KMI of Open University, UK; Differential Ontology Editor from INA France; Domain Ontology Management Environment (DOME) from Btexact technologies; Integrated Ontology Development Environment (IODE) from Ontology Works Inc.; KAON from AIFB Institute and University of Karlsruhe Germany; Protégé 2000

[31] WSML-Rule reasoner: http://tools.deri.org/wsml/rule-reasoner

from Stanford University; SemTalk from Semtation GmbH; WSMT from Innsbruk/DERI; KBE from Vanderbilt University; TopBraid from TopQuadrant USA; and many more.

The author initially selected Protégé 4.0 ontology editor, however, due to some limitations, switched to WSMT ontology editor, both are free, open source, web-downloadable ontology building editors.

2.7 Ontology Support

Ontology is a tool for knowledge representing and elicitation. The ontological representations are engineering artefacts, conflation of related entities in a domain model and an intersection of cognitive science, object oriented concept, lingistics, philosophy and epistemology (Raza and Harrison, 2011c). Ontological enrichment of existing data eliminates latency in the knowledge stream among concerned stake holders and supply chain partners within and across organisational boundaries (Raza and Harrison, 2012a). To automate (fairly) the task of assembly line design and/or reconfiguration, product and resource link points need to be defined at early stages of design and made available easily to be searched, analysed and implemented on 'when and where required' basis (Raza et al, 2009a). To explicitly define these link points, ontologies are used for information capturing and representation, thus integrating PLM systems efficiently into common factory floor information platform. Ontology can be used not only to model the domain of interest but can also serve as a broker to other applications to get the requisite information and to reuse available knowledge of existing and preceding programs.

The existing software applications at Ford are Teamcenter (TC), Process Designer (PD) and recently introduced CCE tool. The CCE tool has just been introduced as a component based tool to support virtual verification of automation systems. The CCE, TC and PD are standalone (operate autonomously), heterogeneous (separate data model and formats) and autonomous (changes are carried out independent of other applications) applications, while the last two are distributed as well. Under these conditions, it is difficult to establish relational information amongst PPR entities as the information is spread across applications. In the current research, the ontologies provided the foundation on which a relational knowledge base with rules and axioms is built where information can be stored centrally in consistent data structures and rules and constraints can be formulated to reduce simulations and manual efforts. As a result, ontologies created a centralised relational knowledge base of Bill of Material (BoM) with its Bill of Process (BoP) and machine Bill of Resource (BoR). Thus ontological based connections and mapping among PPR or in other words, among BoM, BoP and BoR, is formally established and efficiently exploited.

2.7.1. *Alternative Enabling Technologies*

There are many similar technologies for design and implementation of a KB system and this section provides a brief overview of strengths and weaknesses of the similar techniques and the reasons why ontology is given preference over similar technology options to build a KB system. The conventional expert system shells do not allow knowledge modeling and are hard-coded, therefore a more flexible and interoperable tool is needed. This will cover comparison of eXtensible Markup Language (XML), Resource Description Framework (RDF), Resource Description Framework Schema (RDFS) and database conceptual schema with ontologies.

The XML is industry recognised neutral standard to exchange data and create an interoperable infrastructure. XML provides a surface syntax for structured documents, but imposes no semantic constraints on the meaning of these documents[32]. The limitations of XML are addressed by RDF / RDFS. RDF[32] is a collection of triples, each consisting of a subject, a predicate and an object, RDFS adds classes, subclasses and properties to resources, In RDFS, classes can only be named, expressions cannot be constructed to describe other classes (Dickinson, 2009).

An ontology is simply a vocabulary that describes objects and how they relate to one another. One of the main reasons for building an ontology-based application is to use a reasoner to derive additional truths about the concepts you are modeling (Masahiro et al, 2003, Dickinson 2009[33]). OWL adds more vocabulary for describing properties and classes: among others, relations between classes (e.g. disjointness), cardinality (e.g. "exactly one"), equality, richer type of properties, characteristics of properties (e.g. symmetry), and enumerated classes[34,35]. (W3C Documents[36], 2004; Obitko, 2007; Dickinson, 2009).

An important limitation of the ontology competitors (i.e. XML / RDF) is lack of reasoning capability. The ontology, though still not fully matured, was selected after meticulous investigations and analyses and the potential long term benefits compared to its competitors. XML does not has inference and reasoning capabilities and properties cannot be restricted while ontology has got reasoning capabilities and enriched set of properties e.g. cardinality, symmetry

[32] http://www.w3.org/ ; W3C overview documents for XML, RDF/RDFS, OWL etc.

[33] http://jena.sourceforge.net/ontology/

[34] http://www.w3.org/TR/owl-features/

[35] http://www.obitko.com/tutorials/ontologies-semantic-web/ontologies.html

[36] http://www.w3.org/TR/owl-features/

etc. If computer can understand the meaning of the representational language, it can process the information and make inferences by deriving new information. Therefore, many of the basic reasoning tasks may be performed automatically by computer without human intervention and applied in real industrial scenarios and problems. A detailed comparison of XML/RDF/RDFS/Ontologies is available at W3C website. The Table 4 summarises the characteristics of competing technologies to develop a knowledge based system.

	Ease of					
	Interoperability		**Development Cost**	**Domain Modeling**	**Scalability**	**Extendibility**
	Syntactic	Semantic				
Expert system shells	√	x	x	√	√	x
Conventional languages	√	x	x			x
Ontology languages	√	√	√	√	√	√
Standardised Template Driven	√	x	√	x	x	x

Table 4: Comparison of the competing technologies and languages for developing KB system

The ontology provides an infrastructure for standardised representation and efficient management of information. It is used to describe the concepts of real world e.g. a vehicle can be represented differently in different databases but the concept is only one i.e. a vehicle is a vehicle regardless of whether it is being represented for an information system, design / development or for driving, therefore, ontological concepts is closer to human cognitive model of the concepts. In addition to this ontologies provide inherent reasoning to deduce automatic inferences[37] as well as user defined rules and axioms thus a rich level of semantics is obtained. Thus the ontology is used for communication, meaning and context in a certain domain of interest.

[37] Inference is the process by which an ontology or a knowledge based system obtains a new knowledge from already known information.

2.7.2. *Ontology in Automotive Sector*

The use of ontology in the current research and the benefits of inherent reasoning through ontologies is briefly explained here. Considering the example of a car, it has many systems and components, powertrain / engine is one of the most important one. There can be a conventional car or an electric one. The knowledge represented in the ontology takes the following form as depicted in Figure 12.

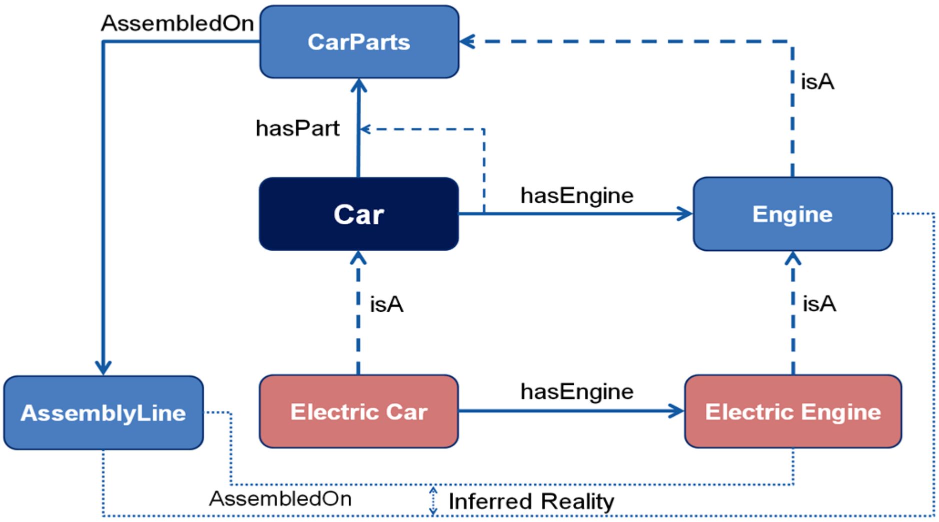


Figure 12: Ontological representation of the 'car having engine' fact

Every car has an engine. A conventional car has an internal combustion (IC) engine or an engine while an electric car is a type of car which has electric engine. Therefore:

Car 'hasEngine' Engine
Electric Car is a Subclass of Car
Electric car has an electric engine.

Therefore if a conventional car is assembled on an assembly line, the sub class of the car i.e. electric car inherits all properties of the parent class hence it is also assembled on an assembly line. This is not defined in the ontology but the ontology inferred this truth by inference mechanism. Similarly in case of automotive assembly line, the benefits of an ontology system can be realised.

2.8 Ontology Driven KB Systems

A constantly growing number of companies that offer professional services or tools in the knowledge management area reflect the use of knowledge based systems for industrial problems

(Davis, 2006). Typical applications of ontologies and rule bases include knowledge and skill management (Staab, 2002) as well as web service and business process management (Fensel et al, 2006). Extensive theoretical studies (Hahn, 2007), some prototypes (Stegmüller, 2003; Küsters, 2006), and projects (Syldatke, 2007) emphasise the relevance of these in the automotive sector. While theoretical issues such as expressiveness and decidability still remain fundamental, non-functional aspects like knowledge base maintainability and security are getting crucial too (Hepp, 2007).

The use of ontology for developing knowledge based systems has been shown by Staab (2001) who used ontology for structuring information and a wider KM infrastructure. Similarly knowledge based system benefits in construction industry are demonstrated by Eldrandaly and Eldin (2006). Xuemei (2007) has investigated manufacturing reconfiguration methodology in multi agent system of reconfigurable assembly line and ontologies have been used for modeling. Ming et al (2003) integrates semantic web knowledge and constructs ontology for internet based manufacturing, provides semantic and reasoning support for intelligent retrieval and discovery of manufacturing resources.

Cao (2003) integrates semantic web and builds manufacturing service ontology to provide capability and semantic information for manufacturing service by decomposing semantic structure of meta-service and simple composite service. Obitko and Marik (2003) have analysed the inefficiency of XML format semantics for describing manufacturing knowledge, and presents a framework that enables to add semantics of OWL ontologies used in the semantic web to the ontologies used in the manufacturing domain to avoid lacking formal explicit description usually expressed in XML format. Tursi et al (2009) described ontological approach for products-centric information system interoperability in networked manufacturing enterprises, managing product related heterogeneous information spread across enterprise taking product as the central object.

For ontology application for manufacturing interoperability, PSL (Process Specification Language) of NIST is typical research. PSL (Gruninger and Menzel, 2003; Gruninger, 2004) is designed to facilitate correct and complete exchange of process information in different manufacturing domain involving scheduling, process modeling, process planning, production planning, manufacturing and machining simulation, project management, workflow, and business process reengineering (Xuemei, 2007).

The vision of TRANSPARENCY[38] is to allow European machine-tool builders for transferring their business into a machine-related cooperation of service oriented partners in a dynamic network. The project aims for a vertical integration of management, design and operation of machine-tools to provide long-ranging transparency for both the end users and the machine-tool builder throughout the whole life-time of the machine-tool. This is achieved by a knowledge based collaborative co-design environment approach using semantic technologies. Special attention is paid for the conceptual system design stage, the feed-back of knowledge from the operational life-cycle stages.

The CommonKADS[39] (Knowledge Acquisition Data System) methodology for the KBS development is a result of the KADS-II project which was a part of ESPRIT 2 (European Strategic Programme for Research and Development in Information Technology) project is the leading methodology to support structured knowledge engineering. CommonKADS offers methods to create coarse-grained descriptions of knowledge-intensive tasks within the overall business process as well as techniques for detailed knowledge analysis, knowledge development and knowledge storage. CommonKADS also provides the methods to perform a detailed analysis of knowledge-intensive tasks and processes. It now is the European de facto standard for knowledge analysis and knowledge-intensive system development, and it has been adopted as a whole or has been partly incorporated in existing methods by many major companies in Europe, as well as in the US and Japan.

Ontoprise[40] is an established German-based organisation specialising in ontology development, processing and semantic applications. Ontoprise has developed a range of solutions primarily based upon ontology engineering. Ontoprise has developed a tool called ‘Semantic Guide’, a generic ontological knowledge based advisory system primarily used for consulting in day to day business processes.

Aletheia[41] – semantic federation of comprehensive product information – is a leading innovation project, sponsored by the Federal Ministry of Education and Research, Germany. Aletheia aims at obtaining comprehensive access to product information through the use of semantic technologies.

[38] www.transparency-project.eu
[39] http://www.sics.se/ktm/projects/kads.html ; http://www.commonkads.uva.nl/frameset-commonkads.html
[40] www.ontoprise.de/
[41] http://www.aletheia-projekt.de/

The approach of the Aletheia project is:

- Gaining knowledge from different sources,
- Uniform representation of the knowledge gained,
- Deducing implicit knowledge,
- Representation of a solution that is adapted to the context and role of the user.

2.9 Commercial Software Applications

This section investigates the supporting technologies commercially available in the automotive sector, demonstrating anticipated benefits, however, there are fundamental limitations when measured from the point of view of knowledge management and technical memory of the organisation. Following are the state-of-the-art software applications being used globally in automotive and aerospace sectors.

2.9.1. DELMIA Automation

DELMIA Automation by Dassault Systemes (DS) is based on recently acquired technology and delivers a next generation collaborative development environment that enables companies to digitally define, control, and monitor automated systems. The advanced virtual commissioning solution provides industrial equipment manufacturers the ability to bring their manufacturing system to life long before it is built. The DS PLM systems is provided specialised solution in automotive product development called Powertrain Engineering and Manufacturing. Dassault Systèmes (DS) Powertrain Engineering & Manufacturing solution consists of applications and industry proven methodologies that address the unique needs of powertrain engineers. It covers the entire spectrum of powertrain development and production activities - from design to analysis and manufacturing (DELMIA World News[42], 2010).

2.9.2. Tecnomatix

Tecnomatix[43] by Siemens offers a range of module applications including Tecnomatix Part Planning and Validation, Tecnomatix Assembly Planning and Validation, Tecnomatix Robotics and Automation Planning, Tecnomatix Plant Design and Optimisation, Tecnomatix Jack etc. Tecnomatix Robotics and Automation Planning enables manufacturers to virtually develop,

[42] http://www.3ds.com/fileadmin/COMPANY/DS-MAGAZINES/DELMIA-WORLD-NEWS/PDF/Delmia-World-News-20.pdf

[43] http://www.plm.automation.siemens.com/en_gb/products/tecnomatix/index.shtml?stc=gbiia400729

simulate and commission robotic and non-robotic manufacturing systems from plants producing dedicated single products to mixed model production facilities with combinations of build variants.

2.9.3. PLM System ‘Teamcenter’ (TC)

A typical use case scenario in UK automotive industry is presented here. TC manages manufacturing / assembly processes’ design within Manufacturing Structure Editor (MSE) module of the tool as shown in Figure 13. The capability of TC in manufacturing / assembly data management does fulfil current industry needs in terms of information management, however, it limits the usability for change management. TC manages products with processes and plant (resources) as static records of data, nevertheless, it fails to provide a constraint-based association which results in manual efforts for any decision making activity.

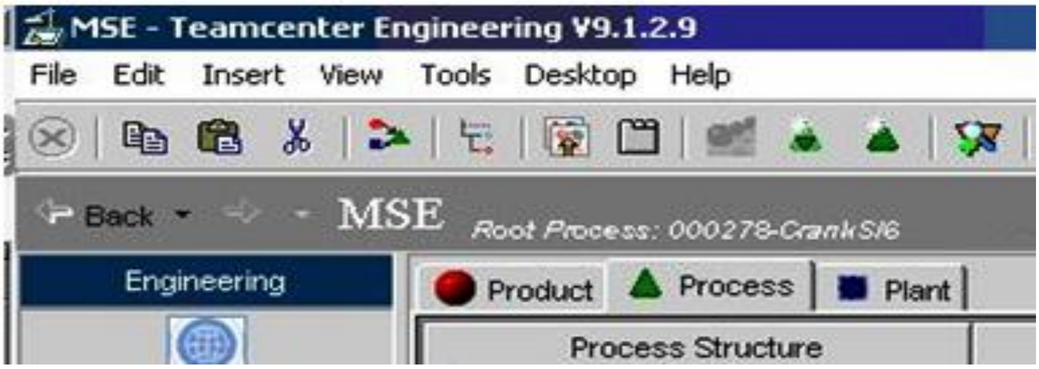


Figure 13: Product, Process and Resource in Teamcenter

- TC arranges BoP structures by relating data in three key areas:
    - ●Product
    - ▲Process
    - ■Plant (Resource/Machine)

Any change in product necessitates checking whether it is possible to assemble the new product on the existing machines. The answer to this question is not a straight forward one. Neither does exist an explicit mapping among products, processes and resources in the present day PLM systems, nor is the capability to define relational constraints in the form of rules and axioms. (Raza and Harrison, 2011b). The current PLM systems are product-focussed and processes are defined as a subset of products. Therefore a separate application, i.e. Process Designer (PD), has to be used in parallel with PLM systems to properly control the key area of process management in assembly systems.

In general, current PLM approaches do not enable product and their under pinning resource systems and associated processes to be readily changed. Whenever there is any change in the product, it is a paramount concern to determine how this change affects associated processes and machines. Without explicit definition of relational knowledge, it is difficult to compare, contrast and critically scrutinise effects of product changes to processes and resources. TC is a valuable tool for information management but does not cope with the industrial requirements of dynamic change management.

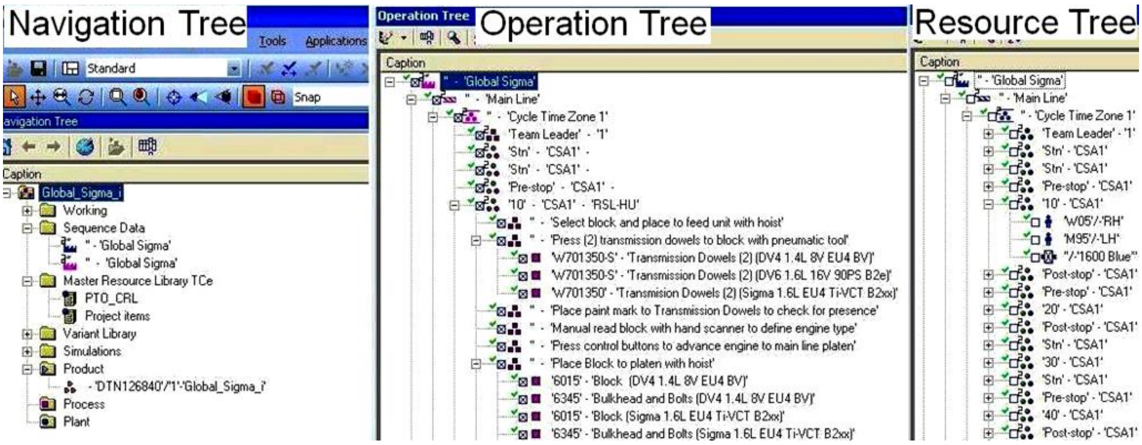
2.9.4. Process Designer

The Siemens Process Designer (PD) is a powerful 3D environment tool for process design, validation, planning and execution. Many of the automotive OEMS use PD for assembly process planning, automotive line builder tool, assembly line design, process modeling and line balancing tool, a major enabler of speed-to-market by allowing manufacturing organisations to bridge product and process design and reduce process planning efforts resulting in faster product launch and higher productivity (Siemens[44], 2010). PD addresses the shortcomings of TC in terms of defining and implementing a complete BoP. PD has three main tabs as shown in Figure 14:

- Navigation Tree
- Operation Tree
- Resource Tree

PD provides navigation capabilities through navigation tree. Process steps are defined for a specific project.

[44] http://www.simplan.de/images/stories/download/Produktblaetter/Siemens_Process_Designer_ENG.pdf?ml=4&mlt=system&tmpl=component



```
Global_Sigma_i
  Working
  Sequence Data
    " - 'Global Sigma'
    " - 'Global Sigma'
  Master Resource Library TCe
    PTO_CRL
    Project items
  Variant Library
  Simulations
  Product
    - 'DTN126840'/'1'-'Global_Sigma_i'
  Process
  Plant
```

```
" - 'Global Sigma'
  " - 'Main Line'
    " - 'Cycle Time Zone 1'
      'Team Leader' - '1'
      'Stn' - 'CSA1' -
      'Stn' - 'CSA1' -
      'Pre-stop' - 'CSA1' -
      '10' - 'CSA1' - 'RSL-HU'
        " - 'Select block and place to feed unit with hoist'
        " - 'Press (2) transmission dowels to block with pneumatic tool'
          'W701350-S' - 'Transmission Dowels (2) (DV4 1.4L 8V EU4 BV)'
          'W701350-S' - 'Transmission Dowels (2) (DV6 1.6L 16V 90PS B2e)'
          'W701350' - 'Transmision Dowels (2) (Sigma 1.6L EU4 Ti-VCT B2xx)'
        " - 'Place paint mark to Transmission Dowels to check for presence'
        " - 'Manual read block with hand scanner to define engine type'
        " - 'Press control buttons to advance engine to main line platen'
        " - 'Place Block to platen with hoist'
          '6015' - 'Block (DV4 1.4L 8V EU4 BV)'
          '6345' - 'Bulkhead and Bolts (DV4 1.4L 8V EU4 BV)'
          '6015' - 'Block (Sigma 1.6L EU4 Ti-VCT B2xx)'
          '6345' - 'Bulkhead and Bolts (Sigma 1.6L EU4 Ti-VCT B2xx)'
```

```
" - 'Global Sigma'
  " - 'Main Line'
    " - 'Cycle Time Zone 1'
      'Team Leader' - '1'
      'Stn' - 'CSA1'
      'Stn' - 'CSA1'
      'Pre-stop' - 'CSA1'
      '10' - 'CSA1'
        'W05'/-'RH'
        'M95'/-'LH'
        "/-'1600 Blue"'
      'Post-stop' - 'CSA1'
      'Pre-stop' - 'CSA1'
      '20' - 'CSA1'
      'Post-stop' - 'CSA1'
      'Stn' - 'CSA1'
      '30' - 'CSA1'
      'Stn' - 'CSA1'
      'Pre-stop' - 'CSA1'
      '40' - 'CSA1'
      'Post-stop' - 'CSA1'
```

Figure 14: Process Designer navigational capabilities

When product is selected in PD, (the assembled product of a specific program) the corresponding lists of BoP and associated BoR can be retrieved. The sequence of assembly process steps cannot be related to the sequence of product assembly as the two may not follow the same order. The resource tree defines workstations associated with their respective process steps and associated parts of product are 'pointed' to resources through TC. This phenomenon is good enough for capturing PPR information and storing at a single place unless the product changes. In case there is any change in the product, the same 'pointed' parts are linked with the already associated process steps and resources without any automatic update in process steps or resources. Therefore, when the user will click on the changed product, the PD would return the previous process steps and resources linked to the product. The current practice is that it is essentially process engineer's responsibility to check, verify, define new process steps and/or recommend potential changes in the resources and finally update the BoP. The suggested changes in the resources are then passed on to the vendors for further investigation and validation which are then again sent back to automotive OEMs for final approval, this may end up having repetitive cycles of interactions between automotive OEM and its vendor before finalisation of the BoR.

One of the objectives of the research is to define assembly processes as relational constraints between product features and machine capabilities. The current PLM/PD data is converted to rich semantic data by adding relationships among the three domains. The 'PLM Product' and 'PD Processes and Resources' are defined as ontological concepts and converted to knowledge elements by adding properties as well as relations with other concepts.

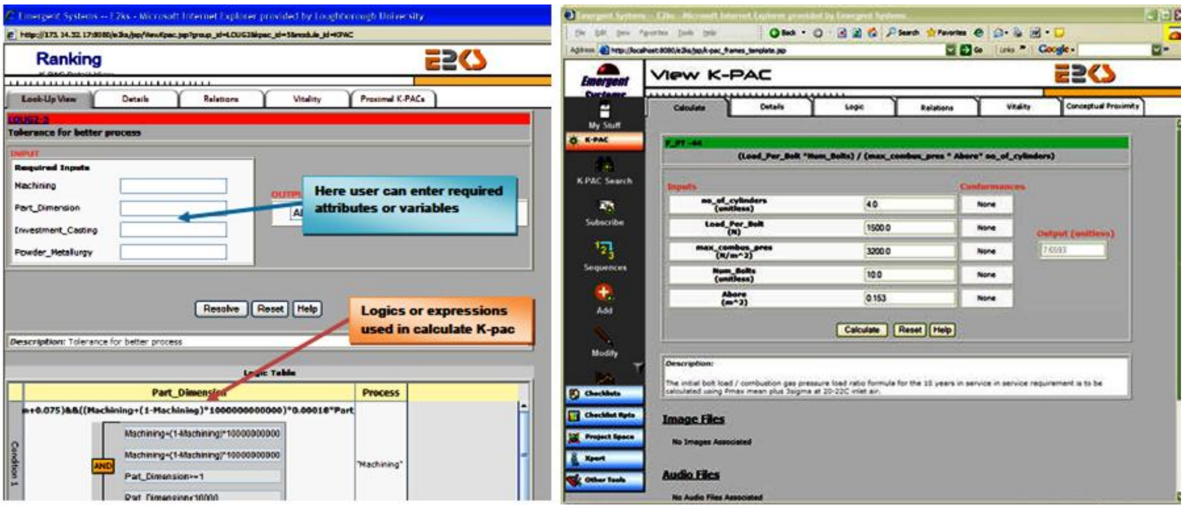
2.9.5. E2KS

E2KS[45] (Enterprise Engineering Knowledge System) by Emergent Systems is a web deployed knowledge management software tool which is used to capture and reuse engineering knowledge. It uses a structured approach to store and manage the engineering know-how. The e2ks helps to build knowledge to form various engineering tools like design guides, engineering assessments, CAD/CAM and manufacturing planning systems and is claimed to be an emerging knowledge management software application. The knowledge is captured through Knowledge Packets called ‘K-Pac’. A K-Pac is a structured representation of engineering knowledge. The K-Pac has distinct advantages over unstructured documents and files when it comes to managing knowledge. There are five distinct types of K-Pacs in e2ks:

- Basic - Defines simple constraints, definitions, and heuristics. Often referred to as 'rules' or 'business rules'.
- Calculate - Defines a mathematical algorithm or expression. Calculate-type K-Pacs allow the calculation to be performed within the K-Pac. For example: A K-Pac can be added to calculate the area of a cube and can be called later with the dimensions of the sides as the input.
- Look-Up - Defines either a property table or a table-driven transfer function. Look-Up K-Pacs work by representing a series of reference values and the relationship of each value to a target property.
- Method - Defines a process that can be represented as a flow chart consisting of steps and logical branches.
- Pic Map - Defines a set of graphics or pictures and the relationship between that set and external data and information.

An example of a ‘Calculate K-Pac’ is depicted in Figure 15.

[45] http://www.emergentsys.com/



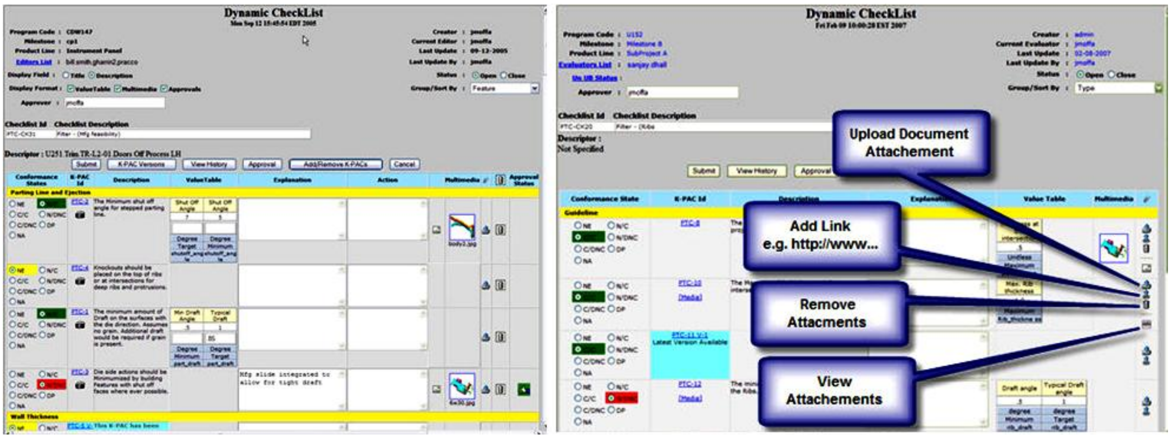
Here user can enter required attributes or variables

Logics or expressions used in calculate K-pac

Figure 15: The calculate K-Pac in e2ks application

The K-Pacs are developed and used for and within a certain Community of Practice (CoP), a CoP is a group of people working in the same domain of interest e.g. A K-Pac can be developed for a modeling CoP and others for machining and so on. Ford is using e2ks for capturing knowledge especially within manufacturing domain. This software application was comprehensively evaluated because initially it was intended to use e2ks for development of an assembly knowledge base. However, there are some fundamental limitations of the e2ks system which restricted the use of it for the current research which are described in the next section.

A checklist is comprised of a set of constituent K-Pacs. To create a tracked checklist, the user must supply information with respect to the project the checklist will track to. The Figure 16 provides a screenshot of the checklist option in e2ks.



Upload Document Attachement

Add Link e.g. http://www...

Remove Attacments

View Attachements

Figure 16: The checklist option in e2ks

The next section provides an insight into the industrial requirements and the available software solutions prevalent in the automotive industry.

2.10 Shortcomings of the Current Software Applications

DELMIA and Tecnomatix are predominantly used for the automation resources design and build process. There are two fundamental issues with these software applications. (1) Both the tools are heavy-weight applications requiring high end computing capabilities with specialised training requirements (2) They are not fundamentally based upon modularity i.e. machines are designed as complete units not from verified modules. Therefore machines become increasingly difficult to amend or reconfigure as product changes progressively occur.

Teamcenter is a PLM system and primarily concerned with product related information management. Though the information about processes and resources can be maintained and mapped yet it is only at the highest level of relevant domain entity i.e. a sub-assembly may be related to a particular station but this information is not linked dynamically. Any change in product would not corresponding update / suggest updating in process steps and resources, therefore, change in any one of the three entities PPR would force updating other entities manually i.e. the current applications lack automatic inference capabilities required for knowledge management.

The e2ks was extensively explored for potential use in the current research, however, there were many drawbacks which restricted the use of it for the current research. Out of the five different types of K-Pac, the two most relevant ones explored are ‘Calculate’ and ‘Look-up’. The Calculate K-Pac defines a mathematical algorithm or expression, it allows the calculation to be performed within the K-Pac, e.g. a K-Pac can be added to calculate the area of a cube and can be called later with the dimensions of the sides as the input. Look-Up K-Pacs work by representing a series of reference values and the relationship of each value to a target property.

Assembly automation involves cross-functional information exchange and it is in this respect where e2ks lacks capabilities. In the current research the K-Pacs can be formulated for product-resource relations. There are two drawbacks with the K-pacs. For a certain assembly stage on a particular station, there can be many axioms and for each axiom category (e.g. reprogram / reconfigure) a separate K-Pac needs to be defined which makes it difficult to manage as the rules are populated. Similarly, two separately defined K-pacs cannot be compared in the current version of the e2ks. This is the fundamental limitation which restricted the implementation of the research in the e2ks application.

In addition to this there are many technical issues which are still being addressed. Most importantly it cannot model information, capture and present a true picture of a domain of interest, it gathers chunks of information and reduces manual calculations. The e2ks defines communities of practices (CoP) for different teams and domains and to get requisite information, K-Pacs need to be pulled through or pushed to a certain CoP. As machine reconfiguration requires interdependent relational constraints, e2ks becomes increasingly complex and it does not remain user-friendly. The e2ks supports knowledge capture and information processing at a basic level (It is primarily a manufacturing-based knowledge management tool) and not suitable for as complex an activity as knowledge management for relational constraints in assembly line design / reconfiguration. It lacks an integrated cross-domain modeling infrastructure. Hence a dedicated software application with information modeling, processing, complex & multiple rules handling and decision support functionalities is required and developed for the current research.

2.11 Assessment and Gap Analysis

The manufacturing industries and especially the automotive sector is facing strategic, tactical and operational challenges. The projects discussed in the last section depict how the operational capabilities of the manufacturing enterprises have been dealt with many research efforts over the last decade or so. It is also observed that the technology has gradually changed from localised company specific applications to generic open architecture techniques, from data modeling to knowledge modeling i.e. ontologies, from data management to information management and then to knowledge management and from stand alone applications to web enabled collaborative application tools. The key success factor for any enterprise is its adaptive capability, the ability to change, innovate and adapt to ever changing and emerging environments. The manufacturing knowledge related projects have emphasised on the following aspects:

- Knowledge based engineering: transfer, share, distribution and protection of knowledge
- Modeling, simulation and virtual tools
- Adaptive enterprises
- Quick and efficient integration of new technologies

Despite considerable developments in the area of reconfigurable assembly systems and knowledge based / expert systems in automotive industry especially for design, development and reconfiguration of automotive assembly lines, there is still a gap in capabilities regarding specific requirements of the automotive OEMs with respect to the required integrative environment especially for the PPR facets. The exhaustive literature review presented in the previous sections suggests that majority of the earlier work in this area is based on planning, optimisation and strategies to achieve optimisation. There are a few similar research efforts that utilise modular techniques, however, they do not consider specific business objectives i.e. a requirement-driven integrative rapid response to the unpredictable product changes. The author considers that a generic framework for management of knowledge, a decision management infrastructure and most importantly a true knowledge management model of the domain (and the lack of these) are the underlying gaps in the current tools and related research efforts.

Similar research studies have focussed the relational association among PPR at a much higher level and since there is no modular tool commercially available / fully functional in automotive industry, hence, no relational linkage down to the component level is practically achievable. As a result no formal axiomatic decision analysis is possible in the prevalent infrastructure void of suitable component based tools. The automotive production sector is a high risk and high investment industry, therefore, strategic alliances are looking at new ways to reconfigure automation systems more readily by adopting new lifecycle support engineering tool and new integration infrastructure with less engineering effort and better business and engineering process management (Haq, 2009). Market segments, product variants and market uncertainties compel automotive sector to come up with new strategies to remain competitive and reduce investment costs. In a perfect world, changes will be confined to the planning stages. However, late changes often occur during construction, and frequently cause serious disruption to the project (Cameron et al, 2004; Harrison et al, 2006). It is demonstrated in the research that powertrain assembly line design / reconfiguration activity can be expedited by providing a knowledge layer along with the existing software infrastructure.

A specially formed committee (that worked under Commission on Engineering and Technical Systems) on Visionary Manufacturing Challenges in USA identified six “grand” challenges for manufacturers that represent gaps between current practices and the vision of manufacturing in 2020 (US National Research Council 1998), and the most important of the six is *‘Instantaneously’ transform information gathered from a vast array of diverse sources into useful knowledge for making effective decisions.* The committee recommended the following key strategic technology and research areas as the most important requirement for meeting the grand challenges (US National Research Council 1998): (1) Adaptable, integrated systems, processes and integrated equipment that can be readily reconfigured; (2) System synthesis, modeling and simulation for all manufacturing processes; (3) Technologies to convert information into knowledge for effective decision making.

The committee recommendations categorically defined reconfiguration, integrated processes, systems and equipment, and knowledge management as breakthrough futuristic technologies. The author's research is directly addressing the said recommendations. The committee also recommended that the most valuable research would be grounded in 'knowledge' of manufacturing strategies, planning and operations because manufacturing is inherently multidisciplinary and involves a complicated mix of people, systems, processes and equipment (US National Research Council, 1998). The author has demonstrated that by implementing a knowledge based system with reasoning techniques in parallel with reconfigurable strategy, assembly line reconfiguration / retrofitting activity can be automated to a reasonable extent.

The lack of axiomatic knowledge / actionable information for the selection and integration of modular assembly automation systems against agreed process steps for assembly of product parts is a major impediment for agility and responsiveness. Assembly automation resources can have several configurations depending upon the inherent design capabilities of the equipment. However, once the resources are configured to a specific assembly operation, the potential capabilities are often overlooked or an effective retrofit is doomed due to the unavailability of the relational constraints among PPR entities. So far no research study has been reported which instead of just advocating modularity as a successful strategy in RASs, practically implements CB technology along with knowledge engineering principles in assembly automation design / reconfiguration.

This research directly addresses the above mentioned gaps in the current methodologies and specifically focuses on an accurate knowledge model to enable integrated view of traditionally separated PPR domains in a formalised specification of the automotive assembly line conceptualisation. There is a lack of infrastructure and tools capable of managing modular components in the engineering domain in taxonomical form and maintaining the linkage of decomposed components down to the smallest functional units in a constraints-based axiom controlled form. The current research is different from the previous efforts because past systems have focused on product optimisation while the focus here is production system optimisation. In particular there is the need for knowledge based support for the virtual engineering of engine assembly automation systems to improve the assembly line design / reconfiguration process.

CHAPTER 3. CURRENT LIMITATIONS AND CONCEIVED SOLUTION

3.1 Introduction

The development and implementation of powertrain assembly line automation resources requires widespread knowledge interactions among several teams using diverse applications to achieve respective goals. A single assembly plant usually assembles a range of engine types and versions. This leads to a corresponding increase in the complexity of the types and sources of knowledge and consequently interoperability problems arise. Therefore, large scale organisations need to have appropriate toolsets and methodologies for capturing, sharing and applying knowledge to help achieve problem solving and decision support activities.

Powertrain designing, though an innovative process, is governed by a few laws. The designer can put most complex and efficient engines on paper but transferring these from paper into practice may pose difficulties. The designer must respect other domains especially machine capability, manufacturing time, general performance, health and safety and overall manufacturing costs. The product designer and machine tool manufacturer must compromise, sooner rather than later, to bring a realistic product into the market. This compromise is always a trade-off between innovation / improvement and reality / practicality and may require a number of alterations both in design of product and machine. After years of experience the product designer eventually starts understanding manufacturer's capabilities, materials feasibility, health and safety regulations and environmental restrictions. However, this still poses a few major threats:

- This is still a human-led activity which may overlook certain aspects
- It is a very complex process usually supported by simulation which is time consuming
- The knowledge remains tacit
- The knowledge may be lost in case of transfer or retirement of expert personnel

The current research is integrating knowledge based and ontologies techniques with the already established reconfigurable systems and component based technology approaches to fill the gap in capability in order to enable more efficient design of assembly automation systems. These techniques are effectively utilised to devise an acceptable, practical system to tackle the stated problem in powertrain assembly automation systems as shown in Figure 17.

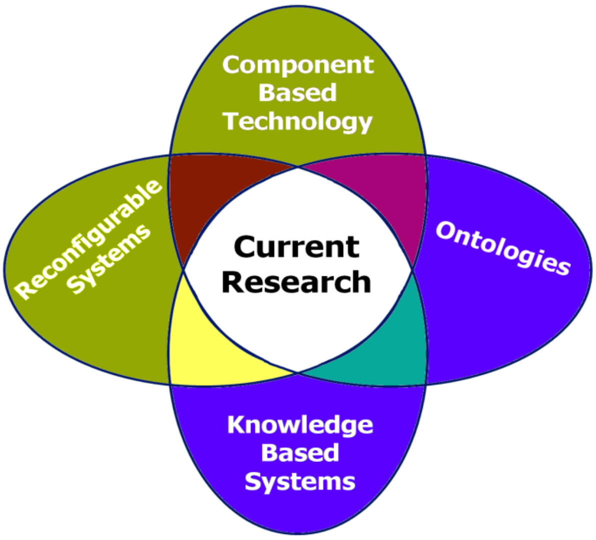


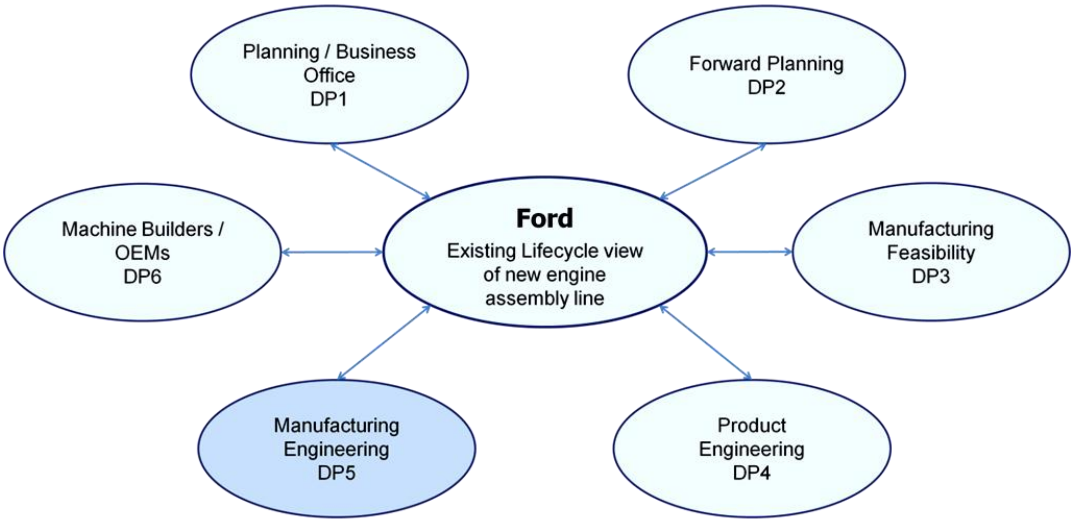
Figure 17: Intersecting themes of the current research

3.2 Limitations in Ford's Business Processes

The Planning and Business Office (DP1) and the Manufacturing Engineering (DP5) are the most important domains / teams from the current research point of view, therefore, a brief of these domains is described here.

3.2.1. Ford Engineering Domain

There are several domains and sub-domains involved in the assembly line design and reconfiguration activity working in collaboration to achieve business objectives as shown below in Figure 18.



Planning / Business Office DP1
Forward Planning DP2
Machine Builders / OEMs DP6
Ford
Existing Lifecycle view of new engine assembly line
Manufacturing Feasibility DP3
Manufacturing Engineering DP5
Product Engineering DP4

Figure 18: The overall context diagram at Ford

Planning and business office is the first domain which develops new business case for a changed product set out by Product Engineering domain (DP4). DP5 and its sub-domains are triggered once DP1 reaches to a milestone called Program Strategy Confirmed (PSC). At PSC, DP1 delivers a new business case formally to the DP5. Machine Builders (DP6) get the targets for mass production through Forward Planning (DP2) and Manufacturing Feasibility (DP3) which is called 1st order to design and build new assembly machines. Currently when the order is initially placed, the *product-resource relational constraints are not evaluated at the end user, hence they have no control over the related engineering processes and in particular the price issues*. The Machine Building (DP6) should be under Ford's control and this research is shifting the responsibilities from entirely on the vendors' shoulders to initial design responsibility to Ford and final build to machine builders. These are the major activities, several intermingled activities have not been taken into account to keep the focus on the scope of the research i.e. mostly on DP5.

3.2.2. Manufacturing Engineering at Ford

Engineering domain and activities among various departments and sections at Ford is again a complex activity with several teams and activities / objectives as shown below in Figure 19.

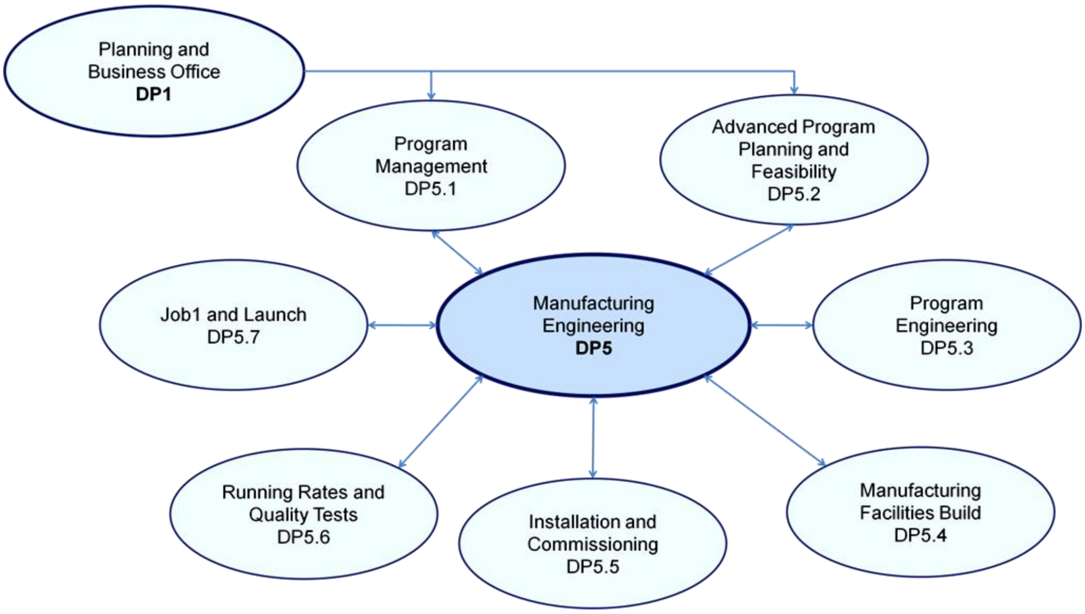

Planning and Business Office DP1
Program Management DP5.1
Advanced Program Planning and Feasibility DP5.2
Job1 and Launch DP5.7
Manufacturing Engineering DP5
Program Engineering DP5.3
Running Rates and Quality Tests DP5.6
Installation and Commissioning DP5.5
Manufacturing Facilities Build DP5.4

Figure 19: Manufacturing engineering at Ford

Planning and business office DP1 is responsible for delivering new business cases in terms of new engine models i.e. new/changed products. Program management DP5.1 is responsible for overall management of the new powertrain business case all the way through the program lifecycle. Advanced program planning and feasibility DP5.2 is responsible for developing detailed production strategies and targets and getting them approved from DP1. Program engineering DP5.3 provides lower level technical specifications i.e. new BoP, BoM and BoR, productivity issues, machine cycle times etc. *These technical specifications are prepared and evaluated manually, therefore, it remains a time consuming activity with potential overlooks and / or postponement of decisions due to shortage of time or unavailability of explicit information*. Manufacturing facilities build DP5.4 is responsible for actual building of new machines against targets set out by DP5.3. Installation and commissioning DP5.5 is responsible for installation, running of machines at production plant along with verifying the machines performance. Running Rate and quality test DP5.6 tries to achieve productivity and quality standards set out by DP5.3. Job1 and launch is the final domain and is responsible to achieve ramp-up period and production launch issues and the new business case is finished by lessons learned activities.

New business case is formally handed over to program management DP5.1 and advance program planning and feasibility DP5.2. The two sections are further supported by a combined team called forward planning and feasibility team, responsible for process planning, overall assembly line design, feasibility and reconfiguration against the changed product. Therefore, the responsibility of the forward planning and feasibility team is to define a new specific Bill of Process (sBoP) from a gBoP. This team works in collaboration with the supply vendors to design and build assembly resources. *However, due to unavailability of knowledge reuse infrastructure, most of the workload is shifted to the vendors.*

The activities and responsibilities discussed above are supported by state-of-the-art IT infrastructure. Product engineering (DP4), Program engineering / Process planning (DP 5.3) and Resource Design/Reconfiguration (DP 5.3 & DP5.4), all three domains depend predominately on PLM system for information management and accomplishing and recording most of their responsibilities and actions. The PLM system consists of a number of modules to help create an environment to integrate PPR domains, at Ford, these consist of Teamcenter (TC) and Process Designer (PD).

A team of experienced engineers from DP 5.1, 5.2 and 5.3, start with existing Bill of Process and generate new / changed BoP by making amendments in the current BoP. The BoP consists of a complete step by step sequence of new operations required for a generic product. The ongoing research project is helping Ford engineers (DP5.1, 5.2 and 5.3) to generate a more pragmatic, reusable and standardised generic Bill of Process (gBoP) by developing manufacturing resource library of pre-defined and pre-validated machine mechanisms. These reusable mechanisms (structural, functional or control) have the ability to be reprogrammed, reconfigured and even redesigned according to changing business requirements. The gBoP becomes the basis for automation assembly design as well as input to other engineering domains. Thus, the gBoP is now more realistic as it is supported by machine modules and components and it helps generating specific Bill of Process (sBoP). After defining the sBoP, the core responsibility of the manufacturing engineering domain is to check how the changed product has affected generic processes and, in turn, will affect the design, build and commissioning of automation resources as well as productivity and tooling issues. It is aimed to create a generic Bill of Resource (gBoR) from gBoP and correspondingly produce sBoR from sBoP. It has already been studied and verified in the current research project that for any change in product, the new assembly line project contains 70-75% commonalities with previous projects (Harrison et al, 2009). *The changed responsibility of DP 5.3 and DP 5.4 will be to evaluate changes required in sBoR derived from sBoP rather than defining sBoR from scratch*.

Though the objectives and targets set by planning and business office DP1 and program management DP5.1 include estimation of productivity and investment costs, yet it is overlooked and deemed to be calculated at later stages. This practice has a huge impact on the program target milestones and results in program delays and cost increase. *Investment costs cannot be premeditated unless products are explicitly mapped to automation resources through processes and unequivocal relationships among PPR are made available at earlier stages of program management.* These unambiguous relations can only be established by providing an infrastructure which interconnects and relates PPR explicitly. As discussed in section 2.10, the current commercially available PLM systems, e.g. Teamcenter, do not have an infrastructure to explicitly map and establish constraint-based relations. Therefore, there is a strong need to design and implement such an infrastructure to fill the missing link of relational constraints and explicit mapping in manufacturing engineering domain. Early confirmation and verification of effects of product changes on the automation resources will have profound consequences on the efficiency and current business processes prevalent at automotive OEMs e.g. the Ford Company. Then mapping of these mechanisms with product and process steps would ensure confident use of results.

3.3 Ford’s Current Assembly Line Design System

3.3.1. PPR Domain Interactions

Engine assembly line design in automotive sector is a complex activity involving many interdisciplinary teams and several software applications. The Product i.e. the assembled powertrain system is designed at ‘Ford USA’, using 3D CAD modeling tool “CATIA” and stored and released in Teamcenter (TC). Once the final approved product is uploaded, engineers at 'Ford UK' access one of the nodes in TC for downloading the newly designed / changed product from TC in USA to TC in UK. The assembly "Bill of Process" (BoP) is used for defining assembly sequence steps for a new product / changed product with the help of "Process Designer" (PD). The assembled product designed in CATIA is then exported into the PD in 'JT' format to define the assembly sequence at shop floor. The 'JT' format (open CAD file format) of 3D models is a heavy weight format and limits simulation with multiple options. The assembly sequence of product assembly in TC is not necessarily the same as the product assembly sequence at the shop floor.

The practical process sequence for the assembly of a complete powertrain system is defined as BoP for a specific engine program in PD. The sequence of process steps defined in PD transforms product parts into product sub-assemblies referred to as 'product-sets' in this thesis, the repetition of this transformation results in the final assembled product. The PD defines process sequence for transforming product parts into product sets by selecting the required product parts from the TC, therefore, the product parts in the TC can be 'referenced' in the PD. The 3D CAD models of the machines are also imported into the PD in 'JT' format. A particular machine carries out a certain set of assembly processes from the finalised BoP. Each product-set is associated with a particular station i.e. resource. It is concluded that the PD defines process steps to be carried out on product parts (product parts get associated with process steps) to convert them to product-sets with the help of resources / workstations (product-sets get associated with resources). A combined use of TC and PD provides information of associated sub-assemblies with respect to respective stations.

There are heavy weight CAD models being used in TC and PD and as the number of workstations against product-sets is increased, it makes the activity slower and slower till the time it takes hours just to upload a few stations which makes it almost impossible to carry out further analysis. There is a tendency to avoid this cumbersome activity unless it is absolutely essential. This forces engineers to investigate the product change effects manually based on experience based judgement. The virtual verification activity for only those stations is carried out which are considered to have potential problems. As stated before, the PD software application requires high processing capabilities, therefore, it becomes a specialised job to be performed on dedicated computing facility with trained staff. The product-resource constraints are evaluated manually on physical resources at the supply vendors. Thus, the automotive OEMS have little control over the design verification and machine building activities. As a result, complete verification of the machines takes place only at the time of commissioning which results in unanticipated problems and time delays in actual commissioning of machines as well as delays in overall program targets. It is concluded that the available software applications cannot cope with the requirements set out by the industrial needs. Before the introduction of the CCE tool, many stages of product build were associated with a single workstation due to the unavailability of the modularity infrastructure within assembly machines. After successful implementation of the CCE tool, a more rational basis of PPR association is available to maintain links among PPR at the component level. A logical next step is to efficiently utilise this modular association among PPR to establish explicit rule-based relations and the current research is addressing the same objective.

3.3.2. Ford's Engineering Process – Simplified

The Figure 20 shows compact form of sequential activities, mainly (but not all) from PPR domains, from a new business case to launch of product at different stages along the assembly line design / reconfiguration activity. A new business case or any change in product initiates a complex cycle of corresponding changes in several domains including assembly machines. As the product specifications are finalised, the corresponding BoM is released. This initial BoM along with finalised product specifications are used for preliminary bid placement. At this point, vendors start evaluating potential machine designs against product specifications while Ford engineers start finalising BoP. Once the BoP is finalised, the same is handed over to the machine supplier and order is placed. The machine builders start finalising specifications of the machines against BoM and BoP. As the simulation based virtual verifications are not exercised therefore there may be revisions in BoP or BoM which may initiate revisions even in product specifications as shown in Figure 20.

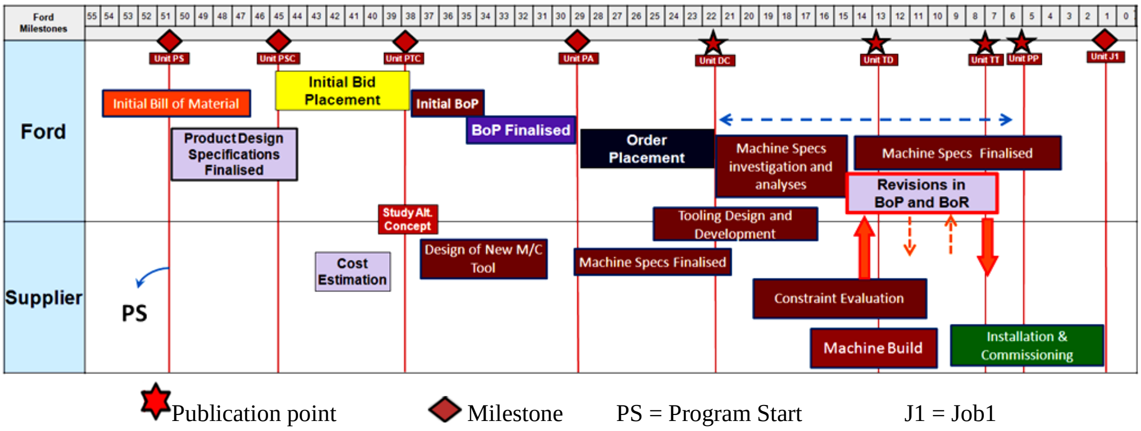


Figure 20: 'As-Is' business engineering process at Ford

The Figure 20 shows that all the major activities regarding assembly resources e.g. design, build, test and commission are carried out by the machine vendors.

3.4 Industrial Need

The continuous change in information, logarithmic growth of data and diversification of decisions along the assembly line design activity requires capturing information along several stages of the activity as well as providing refined and actionable knowledge along several activities. Despite several advanced level applications being used at automotive OEMs (e.g. Ford), there is a lack of knowledge management infrastructure especially for the assembly domain and discrepancy is found in explicit mapping and relational constraints among PPR. Lack of availability of relational constraints results in late approvals of new programs and also dilated time spans from kick-off of a new program to Job1. This is due to the fact that there is no application tool whereby rapid estimation of effects of product change on assembly automation resources could be evaluated. The core requirement, especially after modular based tool implementation, is to develop mapping among PPR so that the product change effects can be readily retrieved. This is suggested to be achieved through a new approach to exercise parameterised analysis of product-resource relational constraints automatically and instantaneously with the help of rules and axioms.

Presently, the Ford engineers have to spend a fair amount of time gathering information from different domains and carrying out manual computations, however, adding ontological knowledge base would help make the process efficient and easy. Addition of semantics would facilitate the information to be understandable by the humans and machines / computer applications. The empirical evaluations performed are summarised in chapter 6, however, a theoretical relevance case is pleaded here. A gap in the market is identified between what is actually required by the industry and what eventually is practiced.

3.4.1. The BDA Project

The current research is part of a wider research project called Business Driven Automation (BDA)[46,47,48,49] project. The project aims to decompose a workstation into smaller useful blocks

[46] http://www.lboro.ac.uk/eng/research/imcrc/research/downloads/211-website-final-report.pdf
[47] http://www.lboro.ac.uk/eng/research/imcrc/publications/IMCRC-2011-Annual-Report-low-res.pdf
[48] http://www.worksmanagement.co.uk/Information-Technology/news/virtual-protoyping-system-could-save-millions-says-loughborough-university/37466/

called modules, this process is repeated till an independent functional unit called a 'component' is obtained. A component is the smallest possible functional unit which can be controlled with the help of logic programs such as PLCs etc. The BDA project outcome has made possible to retrofit the assembly line by reconfiguration and rearrangement of components to construct new workstations for a changed product (Harrison et al, 2009; Raza and Harrison, 2011c).

3.4.1.1 The CCE Tool

The CCE tool is a software application developed at Loughborough University which aids in simulating the virtual commissioning of the assembly machine systems in general and engine assembly line in particular. The CCE tool development was initiated based on the requirements of the automotive industry in terms of large-scale complex automation system design and commissioning. The CCE is a lightweight 3D simulation tool, based upon CB technology, to visualise, test, debug and validate the machine behaviour in a virtual environment. The CCE tool strips an automation resource (workstation) down to smaller functional modules and components and, conversely, builds an automation resource from smaller independent units. With the introduction of the CCE tool for the machine design at Ford, it is possible to evaluate change effects through virtual verification on the affected stations quickly. The CCE tool is a modeling based tool to virtually verify machine behaviour for changed products by altering existing machines or successive joining of verified mechanism components. The tool deals with 3D models, simulation, virtual verification as well as machine control and process logic.

3.4.1.2 Limitation of the CCE Tool

The CCE tool is fundamentally a model-driven modeling / simulation application, hence it lacks knowledge handling capabilities. Though it is a light weight application with reduced complexity of the resources compared to the existing virtual engineering software applications, however, the CCE tool also works in a 3D model-based environment. It has simplified the machine design activity and reduced the time taken for simulation-based virtual verification of product-resource constraints, however, this remains a time consuming manual activity i.e. manually examining the behaviour of every production station. In addition, in common with existing virtual engineering software applications, the results of the virtual verification activity in the CCE tool cannot be recorded in a reusable form. Every time, the product or part of product is changed, the same

[49] http://www.lboro.ac.uk/departments/mm/research/manufacturing-systems/dsg/index.htm

virtual verification will have to be repeated and potential amendments in the machine evaluated manually based on observed simulations.

3.4.1.3 Justification of the Knowledge Based System in BDA Project

Current production machines and associated engineering methods need to be more agile as product changes now occur more frequently and often unexpectedly (Harrison et al, 2006; Harrison et al, 2009). The current level of automation systems is not appropriate when viewed in flexible business context of today's manufacturing systems (Harrison et al, 2009). This may be improved through:

- The provision and use of pre-built, pre-validated, reusable modules to achieve reconfigurable manufacturing systems which are also easily scalable as production volume changes (Harrison et al, 2009);
- Improved access to existing information of products, processes and resources (generic library of machines / modules) by development and implementation of a reusable knowledge base to help start knowledge-driven assembly line management, evaluate the effects of changed products (fairly) automatically instead of engineer's tacit knowledge through addition, removal and reconfiguration of the existing systems in a rapid and cost effective manner (Raza and Harrison, 2011b).

The BDA project approach creates global standard production line configurations (in the form of Gold Standard Library - GSL[50]). When a new product is introduced, its "fit" to each production station must be assessed by virtual modeling and verification of each station systematise the assembly line design process. There is the need to automate this process i.e. automatically advise on the required changes to each station. This need is fulfilled by introducing knowledge layer wrapped around existing software applications.

3.4.1.4 Need for a Knowledge Layer

The new lines / workstations can be cloned from the generic line / workstation. However, there is a missing link to carry out this process smoothly i.e. an upfront knowledge of the changes

[50] The 'Gold Standard Library' abbreviated as 'GSL' is a generic reference library of verified machine mechanisms came into being as a result of initial implementation of the BDA project outcome. The BDA project has successfully created a standardised library of modules and components which can be rearranged, modified, reprogrammed or reconfigured to make a new workstation quickly. The project has suggested creating atleast two generic libraries, one for petrol and one for diesel engines, by complete virtual verification of the whole assembly line.

required in the cloned line / workstation and physical characteristics / properties of the decomposed components of the library in the form of a knowledge based system defining assembly line hierarchical structure, characteristics of the library components, relational constraints and rules, axioms and inference mechanism to produce an integrated readily usable knowledge. The research complements the modular platform by establishing relations among unitary levels of BoM, BoP and BoR.

3.5 Introducing Knowledge Layer

The CCE tool is believed to be a first of its kind design / reconfiguration tool based upon modularity and CB technology. Some of the above stated problems are addressed by the tool, nevertheless, the CCE tool being itself a modeling and simulation based application, cannot cope with the complex information modeling and knowledge management requirements. It is reiterated that the industrial requirement is to rapidly evaluate the product change ripple effects in a more automated manner. To fulfil this requirement, disparate PPR domains need to be brought together, integrated and related. To establish mapping among PPR, information modeling must be carried out which relates domain entities as well as define rules and axioms so that rules and axioms may be defined and used when required.

The author has focussed on improvement of the product change management activity by introducing knowledge layer to design and development of powertrain assembly automation systems. The knowledge modeling and constraint based relational knowledge among PPR facets being the focus to help improve design and development of powertrain assembly systems. The complete automation system design and development is extremely complex and involves a large number of simultaneous engineering activities, this complexity is further increased due to globally distributed supply chain partners (Harrison et al, 2001).

A few important questions have to be answered before any activity can be started in any modular, CB based tool. For example, where to get inputs for the CB based tool and which resource is to model and to which geometrical details? If product or a part of product is changed then which resource or its modules and components are associated with that part or product? The associated modules / components perform or can perform which process steps? What are the resource's capabilities, what processes it carries out or may carry out and which parameters govern dependency constraints to product parts / sub-assemblies, what amendments are required on the existing resource to carry out a similar but deviated operation etc. Subsequently, can the changed assembly is practically possible having modelled successfully in the existing or in CB tool? A part may have become too heavy to be supported by a component of a machine which would have been accurately modelled and simulated and supposed to be fit for purpose. There is no existing solution let alone a readily available answer to these questions. New strategies are required for the manufacturing / automation systems in automotive sector due to the effects of incessant changes in products and consequent changes in processes and resources. This situation becomes worse for automotive industry where the product changes almost every year.

The current prevalent software applications do not explicitly map PPR, though a detailed BoM, BoP and BoR are available, however, this is not available in constraint-based, decision making form. Therefore, the developed system integrates PPR domains and explicitly defines product-process, process-resource and product-resource constraints in order to efficiently manage assembly line design knowledge as depicted in Figure 21.

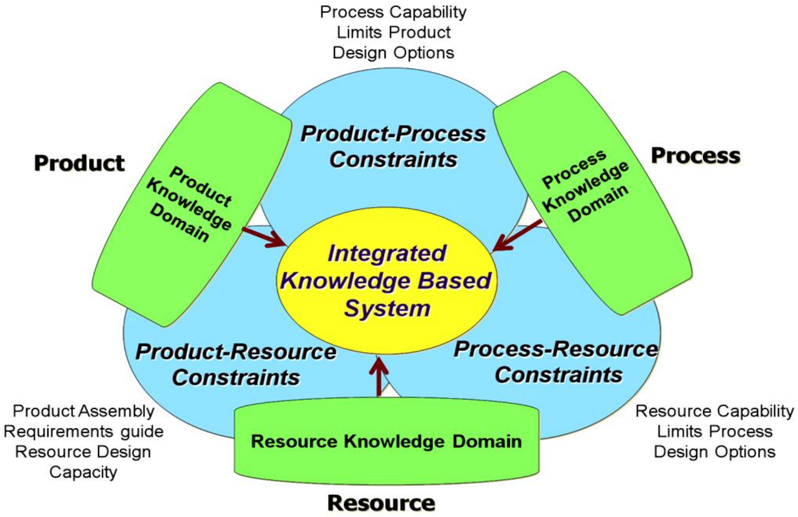

Process Capability Limits Product Design Options
Product
Product-Process Constraints
Process
Product Knowledge Domain
Process Knowledge Domain
Integrated Knowledge Based System
Product-Resource Constraints
Process-Resource Constraints
Product Assembly Requirements guide Resource Design Capacity
Resource Knowledge Domain
Resource Capability Limits Process Design Options
Resource

Figure 21: Relational association and dependency constraints

There is a need for an instant information retrieval of what processes and which resources are linked to different product parts, first and then, how they are related. How the change in product would affect other engineering domains and how the the relational knowledge and mapping among PPR can best be utilised, is presented in this thesis. This ontological knowledge based system can be the basis for a knowledge managemeent infrastructure where rich sources of information and data are effectively used as well as new knowledge is created as the time passes by. The encouraging results provide evidence that KBS is required especially when CB technology is implemented in assembly line designing / retrofitting, the approach is theoretically proved and practically implemented and provides a clear added value to the reconfiguration activities.

The modularity concept for assembly line design / reconfiguration can ease down the complexity matrix of the of machine design, however, the relational connectivity, and in turn, constraint evaluation among the three doamins, increases by the number of modules / components the original entities are broken down to. It is imperative to keep PPR triple connectivity intact regardless of the level of modularity. These components are related to corresponding decomposed assembly operations and product sub-assemblies as revealed in Figure 22.

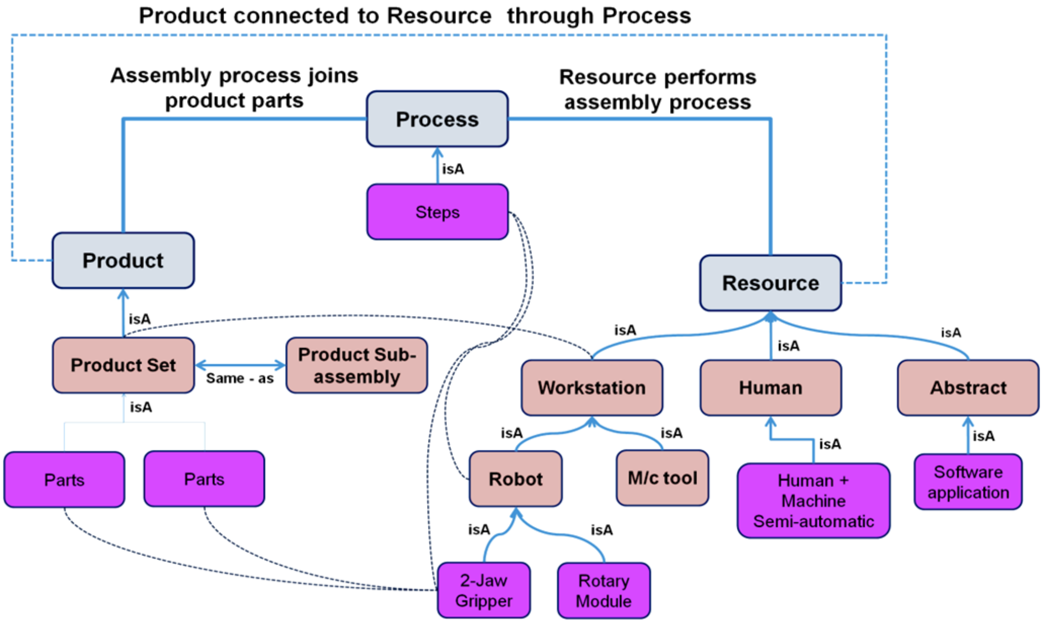


Figure 22: The PPR connectivity and knowledge model add one level down for m/c

By establishing PPR triple association and inter-dependency relationships, speedy changeover of the resources to execute changed / improved processes to manufacture variants of the product can be achieved. Data synchronisation and harmonised representation is the starting point for the processable and actionable information which is the key for creating an integrated knowledge base of products, processes and resources and efficient knowledge reuse in engineering domain of an automotive organisation.

3.6 Summary

This chapter discusses the current practices prevalent at automotive OEMs taking Ford Motor Company as the case for study. An introduction to the BDA project, the CCE tool with its limitations, justification of a KB system along with modular virtual engineering tool and knowledge layer concept is introduced. A simple example of PPR connectivity and knowledge model is explained and information linkage by adding semantics to the data for decision making purposes. Finally a review of current approach and enhancement through BDA project in the form of CCE tool and a KB system is presented.

CHAPTER 4. THE PROPOSED RESEARCH CONCEPT – INTEGRATED KBS FOR ASSEMBLY LINE DESIGN

4.1 Introduction

This chapter provides the proposed research concept of ontological knowledge based support for assembly line design and reconfiguration. The focus of the discussion remains on the ontology and knowledge engineering techniques specifically explored for the new approach and utilised to design and develop the proposed system. The enhancement of the current approach provided through BDA research is extended via the author’s work to the new approach in the form of a complementary KB system.

A review of the existing practices and intended approach is summarised in Table 5.

‘As-Is’ Approach		*Manual evaluation of effects of product change on physical resources*
Characteristics	1	Virtual verification of only potentially problematic stations is carried out as it is extremely time consuming activity
	2	There is no software application to capture the simulation results in a reusable form
‘To-Be’ Approach		*Automatic evaluation of effects of product change on virtual resources*
Basic Technique	1	Virtually verify all the stations with the help of modular and light weight simulation tool (CCE Tool)
	2	A suitable infrastructure to capture the simulation results in the form of rules and axioms and utilise these rules for parameterised evaluation of the product-resource constraints thus reducing, and in some cases eliminating, the virtual verification process against frequent changes in products (My Research)

Table 5: Existing practices vs intended approach

The next section describes the conceptual model to help achieve the ‘To-Be’ approach.

4.2 Generation of the Conceptual and Working Model

This section describes the basic research conceptual model i.e. ontological knowledge based system. The intricate assembly sequence, the sophisticated machine functions and the complicated powertrain parts constitute a challenging industrial scenario, the relational knowledge of which is currently not readily available for a decision making activity. The combination of the three pillars of the engineering domain i.e. PPR and their relational dependency in automotive sector has been the focus of the current research with special emphasis on the powertrain production systems.

4.2.1. The Conceptual Model of the System

The final design of the product drives process planning and process planning drives machine configurations. Hence the product specifications take precedence over BoP and BoP takes precedence over machine design[51], therefore, machine design starts after process plans are finalised and process planning starts after product specifications are completed. Figure 23 illustrates the two main stages, of the improvement process along the assembly line design activity, divided by the dash line. The top portion describes the current status of the engineering activity for assembly line design / build at automotive OEMs, the lower portion describes the outcome of the BDA project i.e. the enhancement of the activity through adopting a modular approach. The lower portion is further sub-divided into two portions, the lower portion shows the outcome of the author's research. The current practices simply associate the PPR entities at the highest level, the BDA project approach defines the PPR entities in terms of CB technology with the same casual relations. The author's research explores the possibility to provide knowledge engineering techniques to explicitly relate the PPR entities down to the smallest units and efficiently utilise the dependency constraints by formulating rules and axioms among the core entities in the assembly line design and reconfiguration activity as shown in Figure 23.

[51] There are two potential approaches (i) standardise the machines and define BoP in accordance with the standardised machines, (ii) standardise the BoP and design the machines according to standardised BoP. Ford typically adopts a position in the middle of these two extremes, however, BoP usually takes precedence over machine design/build.

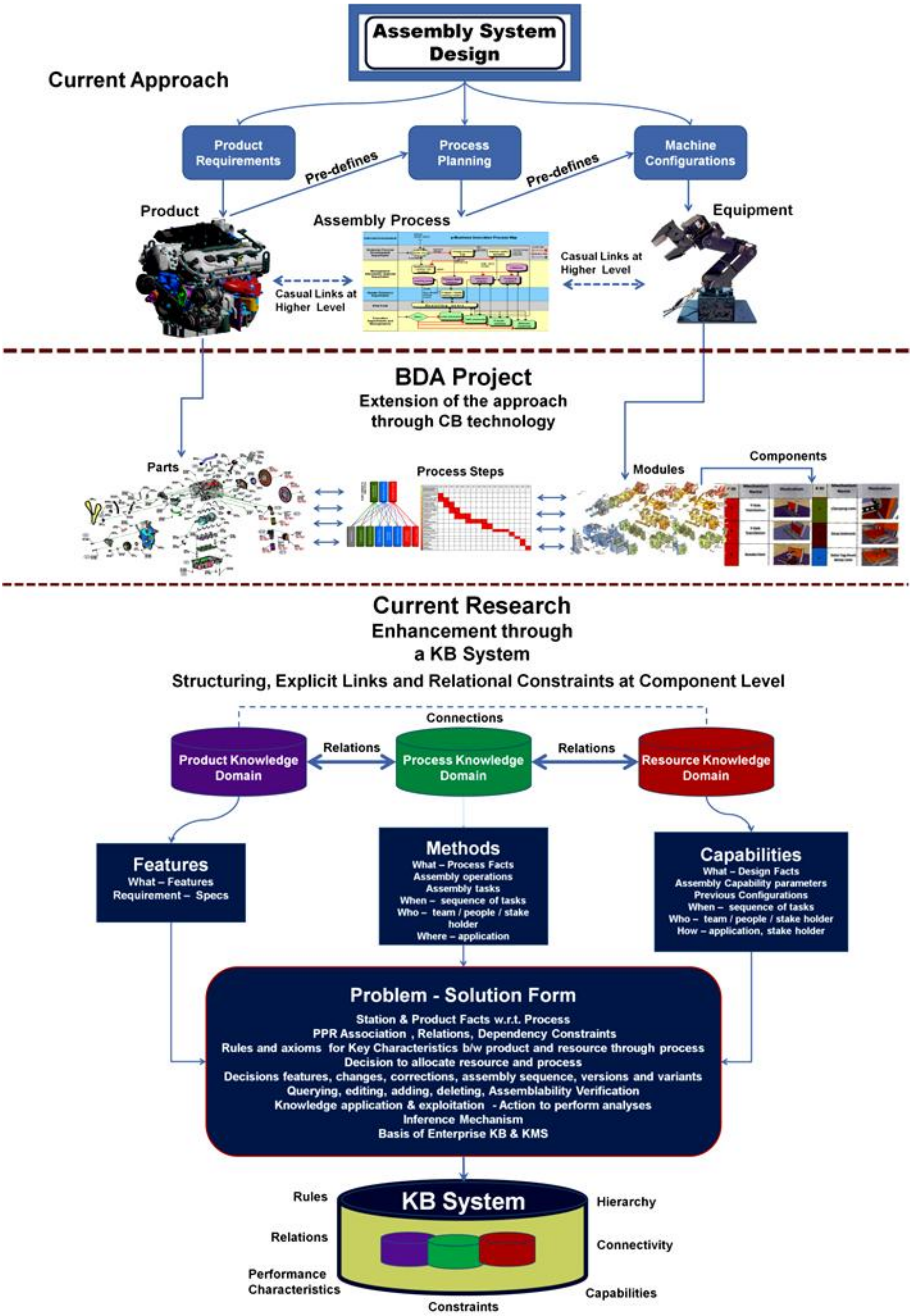

Assembly System Design
Current Approach
Product Requirements
Process Planning
Machine Configurations
Pre-defines
Pre-defines
Product
Assembly Process
Equipment
Casual Links at Higher Level
Casual Links at Higher Level
BDA Project
Extension of the approach through CB technology
Parts
Process Steps
Modules
Components
Current Research
Enhancement through a KB System
Structuring, Explicit Links and Relational Constraints at Component Level
Connections
Relations
Relations
Product Knowledge Domain
Process Knowledge Domain
Resource Knowledge Domain
Features
What – Features
Requirement – Specs
Methods
What – Process Facts
Assembly operations
Assembly tasks
When – sequence of tasks
Who – team / people / stake holder
Where – application
Capabilities
What – Design Facts
Assembly Capability parameters
Previous Configurations
When – sequence of tasks
Who – team / people / stake holder
How – application, stake holder
Problem - Solution Form
Station & Product Facts w.r.t. Process
PPR Association , Relations, Dependency Constraints
Rules and axioms for Key Characteristics b/w product and resource through process
Decision to allocate resource and process
Decisions features, changes, corrections, assembly sequence, versions and variants
Querying, editing, adding, deleting, Assemblability Verification
Knowledge application & exploitation - Action to perform analyses
Inference Mechanism
Basis of Enterprise KB & KMS
KB System
Rules
Relations
Performance Characteristics
Constraints
Hierarchy
Connectivity
Capabilities

Figure 23: The conceptual model of the KB system

The BDA project has provided extension in current best-practices by adopting CB technology especially for the machine modeling activity. The author's research provides further enhancement of the modular approach by establishing explicit cross-functional constraints down to the component level, thus foundations of a new approach to check the machine constraints rapidly and parametrically with respect to the product. As the workstations are decomposed into smaller units, these units must keep link with the parent workstation, the associated product and the assembly process steps as part of a larger knowledge system. The necessity of relational information is multiplied by the number of smaller units created out of a workstation. It is necessary to preserve this information so that whenever new/changed workstations are required to be built out of smaller units, engineers must know the exact function these units perform, their functional properties, limitations and the stage of product assembly they are related to. Therefore, the developed system constitutes an integrated environment composed of relational associations and refinement of concepts through rules and axioms.

4.2.2. The Working Model of the System

The proposed working model consists of usage scenarios and interactions of the developed system with the legacy systems. The user can query (normal or mediated[52]) the KB system for two broad cases:

1. *Efficient retrieval of the 'known facts[53]'*: In this case the user intends to find out the capacity of a machine or retrieve a component / module against a specific assembly operation. The user can send query to the KB system and obtain desired results.
2. *Consultation required for the 'new facts[54]'*: In the case of consultation required to generate knowledge from already known facts and rules, the user can provide additional information as input parameters and the system can provide best consultation for the new facts.

Therefore the developed system may be used for the retrieval of the known facts as well as seeking consultation for the new facts as shown in Figure 24.

[52] Transformation of information in the required context / format.
[53] The information and rules already stored in the KB system, i.e. the 'instance-base' of the KB system.
[54] New information added for logical analysis based upon rules / inferences, also called 'working memory'.

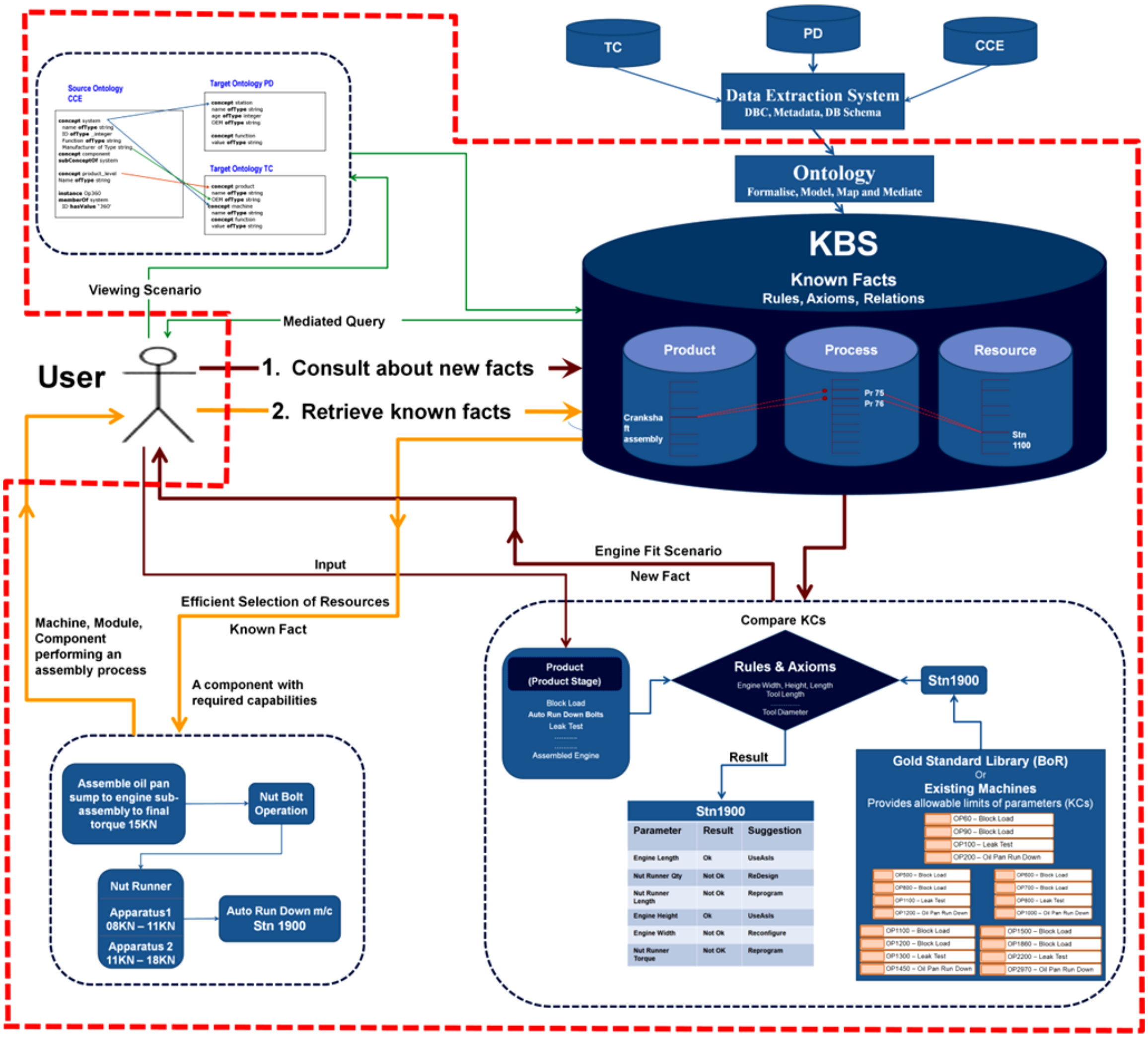


Figure 24: The basic working / implementable model of the KB system

As described before, long-term goal of the BDA approach intends to carry out complete verification of assembly lines through simulation-based analysis, therefore, the complexity of the line design and reconfiguration is reduced and complete verification ensures a confident build and development of workstations on the assembly line. The author’s research complements modular simulation based analysis approach by reducing the virtual modeling and simulation activities through the provision of parameterised analysis of constraints. This aims to ensure rapid assessment of amendments required in the current configuration of the workstations as well as efficient selection of machine units for designing new workstations against changed product specifications. The knowledge based system can access the legacy database systems through a suitable Application Programming Interface (API)[55] designed for the Database Connectivity (DBC)[56], the ontological representation can convert the database concepts, imported in the KB system, into structured representations and store them into the KB system as 'known facts'. The system developer can impose rules and axioms to further refine the defined concepts, where required. The user may either retrieve the instances of the KB system, the 'known facts', or may seek consultation against the 'new facts' as depicted in the Figure 24. Thus, there are two main decision support activities addressed in the developed system for the assembly line design / reconfiguration activity, (i) consultation for changes required in existing (or generic) resources for the changed product, (ii) efficient selection of available resources for the changed product.

For the consultation required for the new / changed product, the Key Characteristics (KCs) of the changed product need to be supplied to the KB system. These KCs will be compared against the rules and axioms defined in the KB system and the output will be provided in the form of decision support for the changes required in the existing resources. The KB system may suggest redesigning a workstation for a certain change in the product and in this case the efficient selection of best available resources can be initiated. The available information in the KB system can further be mediated to resolve syntactical or semantical differences and present a homogenised view of available facts in the form of prevalent practices in a certain community of practice.

4.3 The Integrative Environment: Implementation Steps

An ontology is fundamentally composed of three main elements: a taxonomy of concepts to create a common vocabulary, a set of relationships to link concepts and a set of axioms to control the concept behaviour (Raza et al, 2011e). The basic building blocks of ontology design include:

- Concepts / Classes
- Properties of each concept describing various features and attributes of the concept
- Relations / Restrictions on properties and concepts

[55] API is a set of routines, protocols and specifications used as an interface by software programs / applications to communicate with each other.

[56] A Database Connectivity/ Connection is a facility that allows client software application to communicate with database server software, whether on the same machine or not. A connection is required to send commands and receive answers. For example, ODBC (standard C programming language interface for accessing database management systems DBMS), JDBC (an API for the Java programming language that defines how a client may access a database).

- Axioms

The information of few of the workstations of the engine assembly line at Dagenham powertrain assembly plant was converted into ontology, PPR linked to each other through concepts and properties. The Product is defined in ‘TC ontology’ which represents product ontology, Process in ‘PD ontology’ which represents process ontology and Resource in ‘CCE ontology’ which represents reconfigurable automation resource ontology. Major concepts from the three domains are defined as classes, the classes have the properties as well classes are also related to each other. This is defined at the conceptual / abstract level. The next step is to define real world objects in the form of instances of the defined classes and finally add rules and axioms on the already defined classes and properties. The next section provides detail of the PPR ontologies.

4.3.1. PPR Ontologies

The meta-data of the relevant applications i.e. TC, PD and CCE were defined as concepts in hierarchical form. A brief of the PPR ontologies with example of abstraction and instantiation is described in Table 6.

Product Ontology	Process Ontology	Resource Ontology
Abstract Level (Generalisation)		
• **concept Product** • Name of Type _string • Default Name of Type _string • Product Set of Type (2 *) PP#Product Set • Parts of Type (1 *) TC#Product	• **concept Process** • Name of Type _string • Default Name of Type _string • Description of Type _string • Steps of Type (1 *) _string • Sequence of Type integer • Makes of Type Product_Set	• **concept System** • Name of Type _string • has module of Type (0 *) Module • has component of Type (1 *) Component • Performs of Type PP#Process • has Product of Type Product
Instance Level (Specialisation - Knowledge Base)		
• **instance Product** • Name has value "OP1900_Product" • Default Name has value "Oil Pan" • Product Set has value OP1900_PS • Parts of Type (1 *) TC#Product • **instance** Product_Set • Name hasValue "OP1900_PS" • Parts hasValue {OP1900_Engine_Assy, OP1900_Nuts}	• **instance Process** • Name hasValue " OP1900_Process " • Default name “OilPanRunDown” • Description hasValue "Seals the Oil Pan by inserting bolts" • Makes hasValue OP1900_PS • Steps hasValue { "Clamp adapter plate", "Raise engine into nut_runner spindles", "Run down (15) oil pan bolts to final torque", "Lower engine to roll over height", "Roll over cylinder block (-180°)", "Lower engine", "Unclamp adapter plate", "Write data tag", "Platen leaves station"} • Sequence has value 7	• **instance System** • Name hasValue "OP1900" • Performs hasValue OP1900_Process • has_Component hasValue {OP1900_KitBox, OP1900_Clamp, OP1900_Guarding, OP1900_Lift, OP1900_Timer, OP1900_Sensor, OP1900_Pallet, OP1900_Rail, OP1900_HMI, OP1900_Prestop, OP1900_Rotary_Plate, OP1900_Engine_Plate, OP1900_Nut_Runner} • has_Product hasValue OP1900_Product

Table 6: The PPR Ontologies

Table 6 describes the core of the ontology building process with examples from an actual ontology built during the research. The table shows the concepts from abstract as well as instance level. The three main concepts defined are ‘Product’, ‘Process’ and ‘Resource’. A brief of the major concepts are summarised in Table 7.

Concept	Description
Product	The product is defined as a non-control type of the concept ‘Component’. The product has parts, the parts are combined to make Product_Set by getting parts from TC and assembly process sequence from PD
Process	Process is the sequence of steps and detail of assembly activity which has steps and makes sub-assemblies, the process has steps as defined in PD
Resource	Resource (system / station) is used to carry out assembly operations, it has modules and components as building blocks as well as it should have product parts / product-set to work upon
Product_Set	Product set is the sub-assembly of 02 or more product parts

Table 7: The three main concepts defined in the PPR ontologies

Property	Description
Parts	This is the intrinsic property of the Product, Product has assembled parts i.e. Product-set (1 ∞) AND Product_Set has parts (2 ∞)
Steps	Process has ‘Steps’, The ‘Steps’ are the lowest level of instructions to carry out an assembly bit of an operation. Process must have atleast one step, usually many steps constitute an assembly process. (1 ∞)
Makes	This property links Process with Product. Process ‘Makes’ Product_Set from the product parts
Performs	This property links machine with Process. The System, Module or Component ‘Performs’ process steps to assemble product parts (1 ∞)
has_Product	This property relates ‘Resource’ with the ‘Product’. A system or any of its sub-concepts must have a product to work upon, Resource and Product (1 ∞)
has_Module	A ‘System’ may or may not have a ‘Module’ (0 ∞)
has_Component	A ‘System’ must have atleast 01 ‘Component’ (1 ∞)
Additional Properties defined to get correct Inference model	
requires_System	This property links Machine with its Operation(s) i.e. System to Process
requires_Process	Product_Set requires Process Steps to get assembled This links Product to Process
isAssociatedWith	All sub-concepts are explicitly associated with super concepts e.g Component with Module / System
Extensible KB Properties	
Controls	This property will link tool with embedded level controller

ControlledBy	Inverse property of Controls

Table 8: The description of properties of the concepts in the ontologies

Table 8 provides details of major properties defined in the ontology. The concept ‘product’ is made up of product sub-assemblies / product-sets which in turn are made up of parts. The Product-set has two or more parts of the product. The concept process is simplified as having steps. The (process) steps when executed will carry out assembly i.e. make product-sets. The rationale to the stated decomposition is that an assembly machine is a combination of several independent functional units which can be re-designed/re-arranged to form new machines or reconfigured to accommodate changes in the product i.e. engine.

The reasoning engine concludes automatically that as a ‘part’ and ‘product-set’ have a ‘part/whole’ relation, therefore, every part eventually must become part of a product-set. Similarly every product-set must be formed through a process, therefore, each product-set is eventually linked to a process and so on. If this is deviated from the conceptual level, the reasoner will report error. The hierarchy is defined to clarify and organise the knowledge available on the shop floor level, reduce complexity and build a logical breakdown structure.

4.3.1.1 Illustrative Example – Conceptual Modeling

The station OP1900 is the automatic assembly machine which tightens the bolts on the oil pan sump of the latest Ford Fox engine. The system / station ‘Auto Run Down’ and its ontology are shown in Figure 25.

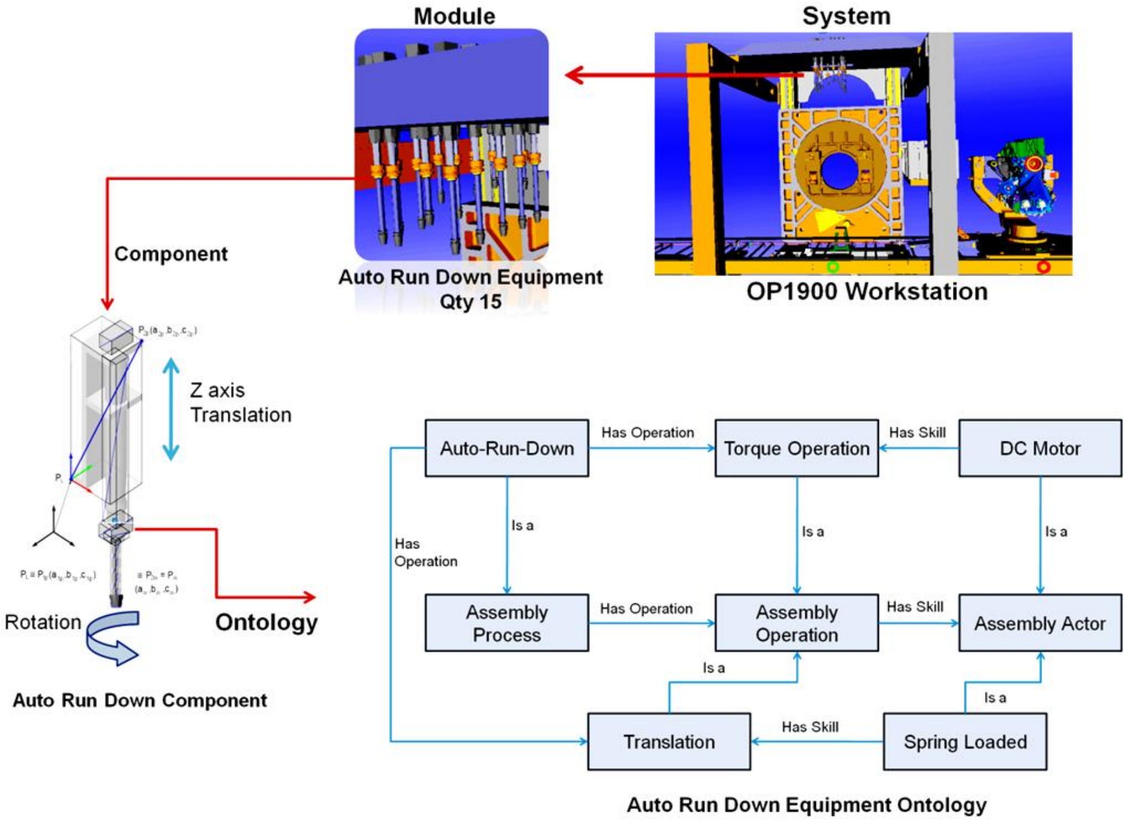


Figure 25: The 'Auto Run Down' equipment and its ontology

Figure 25 shows the process of converting equipment functional characteristics into an ontological representation. The 'auto run down' equipment is used to tighten nuts on the bolts on the OP1900 station at Ford's Dagenham plant. The 'auto run down' equipment is a module on OP1900 and consists of 15 identical components assembled together in the required configuration for the tightening of bolts. A single 'auto run down' unit is the basic functional unit i.e. a reusable component. As shown in Figure 25, it has two operations i.e. translation and rotation / torque application. The translation is provided through spring loaded mechanism while rotation is provided through a DC motor. The ontological representations formally define the functional characteristics of equipment and relate them to the assembly process.

Product is connected to resource through possible process steps. The developed system provides support in assembly line design and reconfiguration through establishing relations among PPR from top down to component level. This has proved reduction of efforts and time during the line design / reconfiguration activity. The benefits of the modular reconfigurable tool can be fully realised through a knowledge based decision making support which then can be connected to legacy systems to extract information and apply reasoning / rules and axioms thus adding a knowledge layer. Thus the developed system helps realising the vision of readily adapting the assembly line / automation resources to the changing market demands through rapid evaluation of constraints among PPR elements. This is realised through explicit relations defined among the PPR entities.

4.3.2. PPR Mapping

The previous section has described the generation of PPR ontologies. This section describes how mapping among the three domains i.e. PPR is carried out in the ontologies. Generally there is no explicit mapping available in the unambiguous and problem-solving form in the automotive sector because the PPR entities are dealt by autonomous teams with independent goals. The explicit mapping is defined in the ontologies through (i) properties of the concepts and (ii) relations among the concepts. For example, the concept ‘System’ in CCE (Resource) ontology has had the properties “has_Product”, with attribute product / product parts and “has_Process”, with attribute process / process steps. The attributes are then retrieved from the respective ontologies i.e. from ‘TC’ (Product) and ‘PD’ (Process) ontologies respectively. In this way, the PPR entities are mutually mapped to each other, as a result, all the instances of the concept ‘System’ (workstation) will be related to specific product and specific process steps; processes to products and resources and resources / machines to products and processes. Hence the three separate domains have been linked together as shown below in Figure 26.

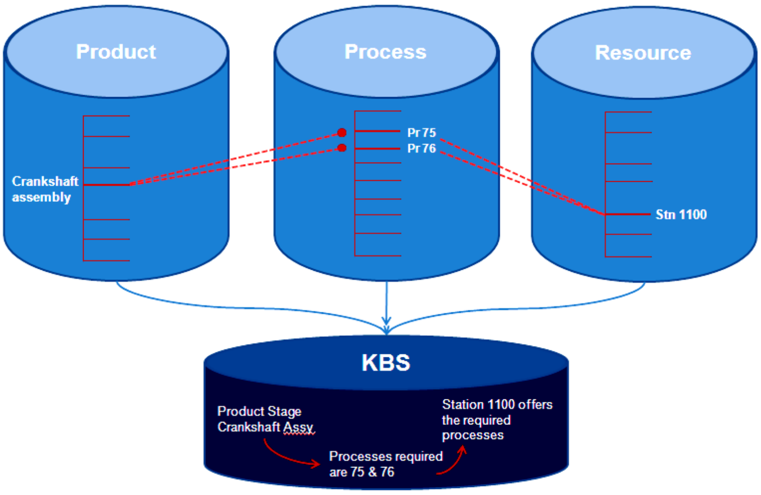

Product
Process
Resource
Crankshaft assembly
Pr 75
Pr 76
Stn 1100
KBS
Product Stage Crankshaft Assy.
Processes required are 75 & 76
Station 1100 offers the required processes

Figure 26: PPR mapping in the KB system

Once the concepts in the ontology are defined and relationships among PPR established at the conceptual level, the next requirement is to utilise these mappings at the instance level i.e. real world objects. For example, the product (product_stage) at OP1900 is to be mapped to the respective automation resource and process steps. If the relations are defined accurately at the abstract level, it is easy to duplicate the same relations at concrete level in the same consistent manner. For example, in Figure 26, the instance of the resource 'Stn1100' performs process steps 75 & 76 to assemble the product-stage called the 'crankshaft assembly'. Therefore, the mapping at the concrete instance level produces relations which produce the integrated PPR environment as shown below in Table 9:

Product	Process	Resource
If 'Product_Set' *has* 'Process' *And* Engine sub-assembly *is a* 'Product_Set' ***THEN*** Engine sub-assembly *has* 'Process' (Oil_Pan_RunDown)	***If*** 'Process' *makes* sub-assemblies *And* 'Oil_Pan_RunDown' *is a* Process ***THEN*** 'Oil_Pan_RunDown' 'Process' makes a sub-assembly by adding 'oil pan sump' through sealant and 'nut bolts' to (existing) engine sub-assembly	***If*** {'Resource' *performs* 'Process steps' AND *If* 'Resource' has 'Product'} *And* OP1900 System *is a* 'Resource' ***THEN*** 'OP1900 Station' carries out (Oil_Pan_RunDown) 'Process Steps' on (OP1900_Engine sub-assy and nuts-bolts)
Derived from relational conceptual model	*Derived from* Conceptual and Instance-level definition	*Derived from* relational conceptual model & Instance-level definition

Table 9: PPR mapping through relational binding and reasoning

The mapping is facilitated through a template which bounds the input values in consistent to the abstract level relationships. The user of the system cannot omit / skip abstract level rules while filling in real world objects as the template will force to provide complete and consistent information. If there is any contradiction to the abstract level rules, the ontology will report inconsistency error.

4.3.3. Relational Constraints

The previous two sections have explained the process of defining basic concepts in the ontology and establishing mapping among the concepts. This section describes the procedure to define relational constraints among the already defined and related entities in the ontologies. Complex issues arise in defining relational constraints, many of which are neither clearly defined nor completely resolved during the simulation evaluation. The interrelated constraints form complex spatiotemporal relationships among PPR entities, therefore, form the bedrock for the current research study. The interdisciplinary constraints among PPR entities depend upon spatial characterisation and change successively on each station as the product parts keeps assembling as per defined process steps, therefore, are a function of temporal correlation.

The relational constraints are defined in the form of rules and axioms among the already defined concepts in the ontologies, hence, the relational constraints are axiomatic constraints. The rules and axioms in the ontologies require some prerequisites i.e. clearly defined concepts, consistent structure of the hierarchy and relations among concepts established and mapped. This is one of the breakthrough pieces of work added in the Ford library of knowledge. Presently, the relational constraints are evaluated with the help of 3D modeling / simulation software applications or through Virtual Build Events (VBE[57]) which are expensive, time consuming and laborious tasks / events as well as there is no formal mechanism to document the results. The research directly addresses these issues.

The process starts with defining Key Characteristics (KCs) of the 'product stage' on a specific station. The 'product stage' is the sub-assembly of the product on a specific station during the course of assembling.

Defining Key Characteristics

Key characteristics are the dimensional constraints on a particular workstation which govern successful assemblability of the product parts or otherwise. For example, few of the typical KCs are tooling characteristics, quantity of tooling required, length / diameter of tooling, clearance space in X (engine length), clearance space in Y (engine width), torque / assembly allowances, clearance space in Z (engine height) and product weight.

Defining Rules and Axioms

Once KCs are defined for a given workstation against a specific product assembly level, the next step is to define rules and axioms which govern the possibility of a range of variants of product assemblies on a particular station. Rules and axioms further impose restrictions on the concepts under particular conditions. The axiom is defined in terms of key characteristics of the product / workstation against a specific assembly level of the product. The actual values of the KCs for the changed product are compared with the existing parameters of the automation resource and the result is displayed as one of the four possible options for the automation resource i.e. Use-as-is, Reprogram, Reconfigure, Redesign. Therefore, the rules on the KCs are used to quickly evaluate where the changed product falls under one of the four categories.

[57] An investigative event involving cross-functional teams to review the product, manufacturing / assembly process and potential assembly hard points with respect to the automation resources.

Defining rules and axioms is not a straight forward task. This phase of the ontology involves extensive analysis of the automation resources w.r.t. corresponding product stage. The knowledge is implicit based on experience and skill of engineers and a detailed analysis ensures selection of correct KCs as well as precision and suitability of the rules. The axiomatic constraints vary from station to station however the author has formulated a generic guideline for the selection of four categories of the automation resource described above.

4.3.3.1 Four Categories of Change

One of the characteristics of the ALDIMS tool is unambiguous classification of the possible change types required against new / variants of existing product. The four categories of retrofit defined in the KB system are described in Table 10.

S. No.	Category	Explanation
1	Use-as-is	The product change doesn't affect the existing resource therefore same resource may be used for the changed / new product
2	Reprogram	The product change affects the set-up of the resource, however, the required capability is within available capacity of the resource which needs to be amended through software adjustment
3	Reconfigure	The product change affects the resource in such a way that the current configuration needs to be adjusted, the required capability is not possible with the current configuration, however, the machine may be readjusted with addition / replacement / enahancement of one or modules / components i.e. hardware and software adjustment
4	Redesign	The product change has a major effect on the resource and many of the modules / components need reconfiguration or redesign therefore the resource needs to be redesigned

Table 10: PPR mapping through relational binding and reasoning

These change categories are consistent with industrial practice and provide a readily available action plan for the process and manufacturing engineers as well as a much deeper insight for domain experts and programme managers.

4.3.3.2 Broader Rules for Line Design for a Changed Product

Define rules/axioms in ontology for PPR relationship i.e. which station / systems are related to product / process (mapping) and how they are related (effects of changes through rules). A broader set of rules on top of the specific ones are formulated as described below:

- If the values of the KCs for the changed product are exactly the same as the existing product then it will be used as it is.

- When some values of the KCs change which do not affect configuration of the system i.e. tool position and product positional reference remains the same but features of the product change then it usually falls under the reprogram category.
- When a feature of the product changes in such a way that it loses its location with respect to the previous product while the overall space limitations of station / tool or product are within prescribed limits and fulfilled then it will be reconfigured.
- When the overall space limitations of either the product or the station / tool are affected then it will be redesigned.

Based upon the above stated observations, the following guidelines are set out for defining rules and axioms as well as selecting the correct category of the changed resource.

- Product has same feature and same location – Use-as-is
- Product has similar feature (changed dimensions) same location – Reprogram
- Product has same but offset feature – Reprogram or Reconfigure
- Product has a changed feature same location – Reconfigure / Redesign
- Product has a changed / different feature and changed / different location – Redesign

An assembly process has many facets e.g. safety, ergonomics, function, flexibility, capacity, lifecycle cost, etc. however, initially only functional behaviour is considered. Similarly machine / station has numerous characteristics, however, the emphasis is on overall assembly line design/reconfiguration activity and only those aspects are considered which directly affect the PPR triple. This will reduce the revision time spent for comparative fit-analysis between a new engine variant and its production machines.

The rules may be formulated through one or combination of the following methods / techniques:

- Digital Pre-Assembly, i.e. simulating the 3D models of existing product-resource,
- Virtual Build Event, i.e. manual assessment of the physical parts and required process.
- Slow Build Event, i.e. relying on the judgement of the process engineer.
- Human tacit knowledge.

The rules and axioms defined in the example were designed after a thorough simulation activity.

Example:

The developed system contains the application of rules of the type:

IF	condition	A, B, C exist
THEN	conclusion	D is true
ELSE	conclusion	E is true.

The '*If-Then-Else*' types of rules are also called production rules.

Consider a system which performs an assembly operation of tightening bolts (e.g. on the station 1900 at the Ford assembly plant). In this case, the diameter of the bolts and length of the bolts are the key characteristics of product. Similarly the key characteristics of system / station are torque to be applied, distance travelled in Y-axis to tighten the bolts etc. The KCs and their attributes are summarised in Table 11.

Resource Key Characteristics		**Product Key Characteristics**			
		Existing		**New**	
Attribute	*Value*	*Attribute*	*Value*	*Attribute*	*Value*
Distance possible in Y axis	50mm	Diameter of bolt	10 mm	Diameter of bolt	16 mm
Max. torque possible	50Nm	Length of bolt	25 mm	Length of bolt	35 mm
Width handling capacity	75mm	Torque	45 Nm	Torque	55 Nm
Length handling capacity	50 mm				
Diameter handling capacity	14mm				
Results					
Diameter of the nut runner		Redesign (new set of nut-bolts)			
Length must be in range		Use-as-is			
Torque must be in range		Reconfigure			

Table 11: Simplified example of the KCs of Product vs Resource on OP1900 station

Now if the product changes, then first the system or components of the system affected directly with this change is to be retrieved, and then it is required to check if the changed product can be assembled on the existing resource. For this, KCs of the product and the system will be compared. The KCs of the changed product will be compared by matching parameters against capabilities of the station. If the parameters of the changed product fall outside the resource capability then it is not possible to assemble the product on the resource in the current configuration state. As a result, the resource needs to be reprogrammed, reconfigured or redesigned. In the current example in Table 11, the station needs to be reconfigured because the maximum force / torque possible through the existing station is 50Nm and the changed nut-bolt system requires 55Nm. Therefore, a new nut runner tool may replace the existing one. A simple rule can be added in the KB system e.g. if torque required is greater than 50Nm then the station needs to be reconfigured. In this way, we can check all the defined KCs and define simple rules accordingly.

Rule

Converting expert knowledge into rules:

If Torque > 50Nm
Then Reconfigure the Component (on the Workstation).

The above rule is for a single characteristic of one part of the product. In actual scenario, there are usually many KCs inter-connected with each other and with the automation resource, therefore, all the KCs are defined and evaluated through rules and axioms. The rules on a single station may become quite complicated, however, the effort of formulating rules will benefit for the rest of the programs on a particular assembly plant as well as for new programs providing a reference starting point. Again a real but simplified example of a rule defining whether a product can be assembled on a particular resource, defined in the ontological KB system (in WSML-Flight format) is shown below:

Axiom Station 1900

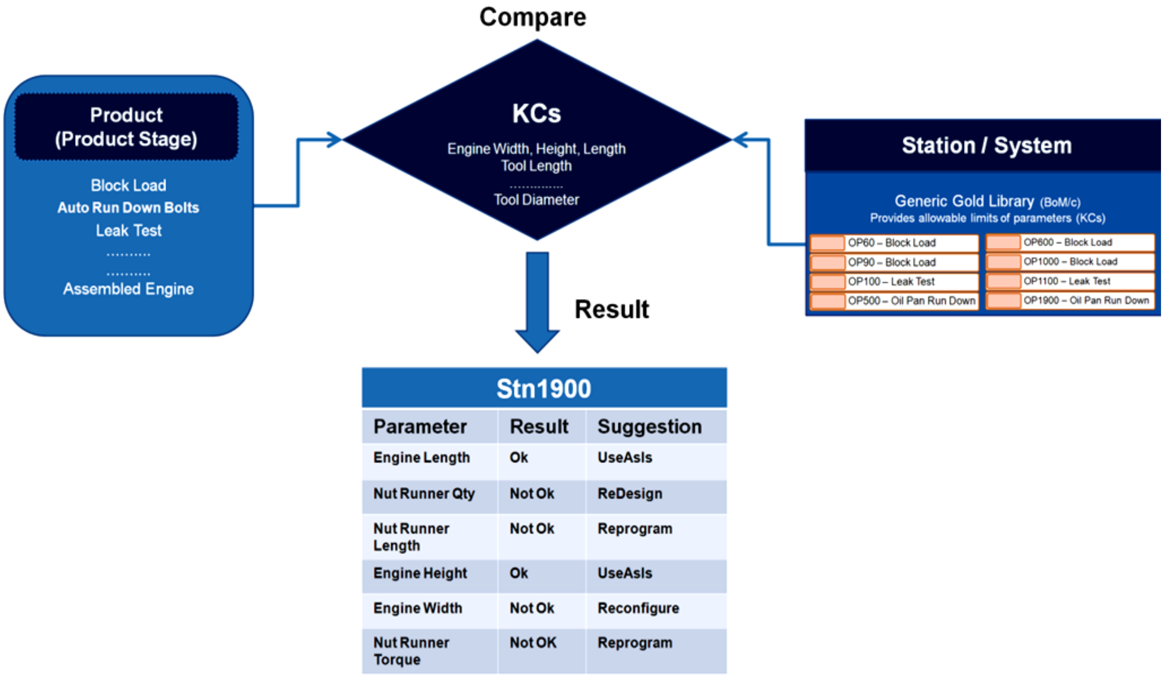
Defined by ?x member of Product

If Product_Length **<** Station X-axis max available space **and**
Product _Width **<** Station Y-axis max available space **and**
Product_Height **<** Station Z-axis max available space **and**
Diameter & Tool travel required **≤** System tooling capability **and**
Force / Torque required **≤** System tooling capability **and**
Product_Weight **≤** Maximum allowable weight on station
Then Implies ?x member of Station 10.

The above axiom is defined in the WSML ontology language. The general logical expression syntax for WSML-Flight is predominantly based on logic programming style, in the sense that it has constants, function symbols, variables, predicates and the usual logical connectives.

4.3.4. Comparison of the Key Characteristics

The rules and axioms defined in the PPR ontologies are based upon comparison criteria. The axioms compare KCs of products with respect to machines. As the product changes at every station (more parts are assembled at each station) therefore, it is necessary to define a certain product build stage / product assembly level which should be compared with the respective station. Product build stage is defined as an integer, therefore, product build stage1 will be on station1, stage2 will be on station2 and so on. If key characteristics of a specific assembly stage of the product are compared with a different station, it will be incorrect and would not give correct results. For example, to check whether the changed product can be assembled on station1900, it is necessary first to retrieve the product for that stage and then compare the KCs of product vs machine for that stage defined in the KB system as depicted in Figure 27.



Parameter	Result	Suggestion
Engine Length	Ok	UseAsIs
Nut Runner Qty	Not Ok	ReDesign
Nut Runner Length	Not Ok	Reprogram
Engine Height	Ok	UseAsIs
Engine Width	Not Ok	Reconfigure
Nut Runner Torque	Not OK	Reprogram

Figure 27: Compare KCs of resource against respective product build stage

The KCs of the correct product stage are compared with the functional limitations of the respective automation resource through the defined rules and axioms and the output is used as a decision guide for reconfiguration in the individual automation resources and subsequently in the complete line.

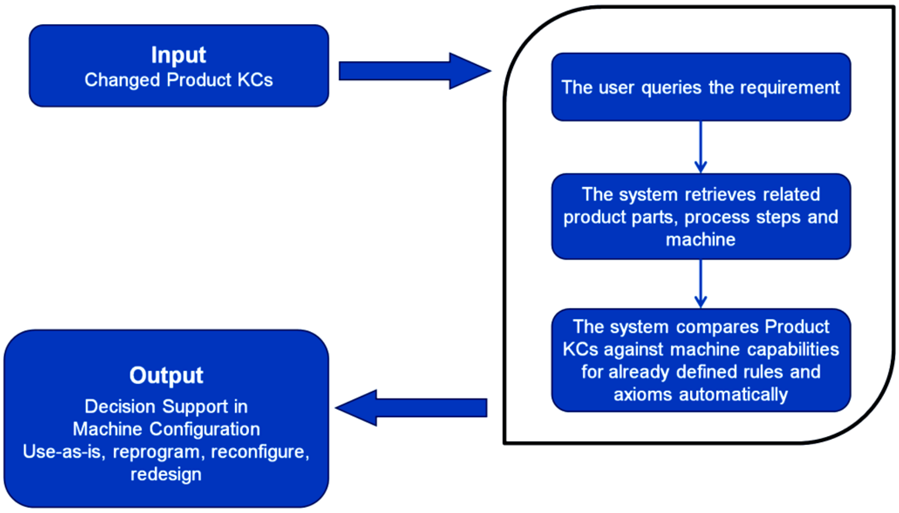

Input
Changed Product KCs
The user queries the requirement
The system retrieves related product parts, process steps and machine
The system compares Product KCs against machine capabilities for already defined rules and axioms automatically
Output
Decision Support in Machine Configuration
Use-as-is, reprogram, reconfigure, redesign

Figure 28: Summary of steps for quick constraint evaluation

The steps are summarised in Figure 28. The changed product requirements are to be compared with the existing station capabilities. Assembly processes provide the product build stage and comparison of the correct assembly stage of the product with corresponding stations is accomplished in the developed system.

4.3.5. Summary of the Potential Effects of the Research

Table 12 summarises the effects of the BDA project and the current research:

Powertrain Assembly Line Lifecycle Activities			**As-Is**	**BDA**[58]	**ALDIMS**
1	Define and finalise BoP		Manual	Manual	Semi-automatic
2	Map product build stage to BoP		Manual	Manual	
3	PPR Mapping		Manual	Manual	Automatic
4	Virtual Verification	Machine design	Vendors	Manual *(Automotive OEM)*	Semi-automatic *(Automotive OEM)*
		Constrain evaluation	Vendors Manual	Manual *(Automotive OEM)*	Automatic *(Automotive OEM)*
5	Rules and Axioms		Non-existent	Non-existent	Available
6	Machine Build		Vendor	Vendor	Vendor
7	Decision Support in M/c Reconfiguration / Redesign		Not available	Not available	Available
8	Decision Recording in reusable form		Not available	Not available	Available
Support Activities					
9	Efficient selection of resources to redesign (or reconfigure) resources		Manual	Semi-automatic	Automatic
10	Data mediation		Not available	Not available	Available
11	Hierarchical classification of machines		Not available	Available but Limited	Available
12	Cost estimation & analysis		Human judgement	Not available	Future work Planned to link categories of change to cost functions

Table 12: Summary of the effects of the BDA and ALDIMS on the existing approach

As the present automotive sector works beyond geographical boundaries and has to share knowledge globally, therefore, the proposed system is also accessible over the world wide web with the accessible rights given. The designed and developed KB system provides an integrated PPR knowledge management tool so that the user can select, search, edit or query required information as well as navigate through the KB system. In developing ontological knowledge based system, two well known approaches are considered i.e. (1) the standardisation approach, wherein all are encouraged to use a common, shared, standardised ontology terms, (2) the

[58] The BDA approach involves the use of virtual engineering tools at the end user for process planning and machine layout

mediation approach, wherein ontology is used as an interpreter/mediator among different terms. The current system endeavours to standardise the vocabulary, however, mediation approach is also used, where required.

4.4 System Description

In the developed KB system, formal knowledge representation is achieved through ontology. The ontological KB system is used to bring formal knowledge representation to the PPR domain. Knowledge is the focus of the system and acquiring knowledge from existing applications and domain experts and then structuring it to be shared and reused is a critical part of the system. A multi-user communication decision support workbench is designed and developed. The distinguishing feature of the KB system is the semantic enrichment of the existing information by providing the right context for communication and sharing. The basic architecture of the designed system consists of:

- Object Oriented Design
- Logic Programming and First Order Logic

The adaptation of existing data structures and an extendable architecture with guidelines and correct specifications is achieved.

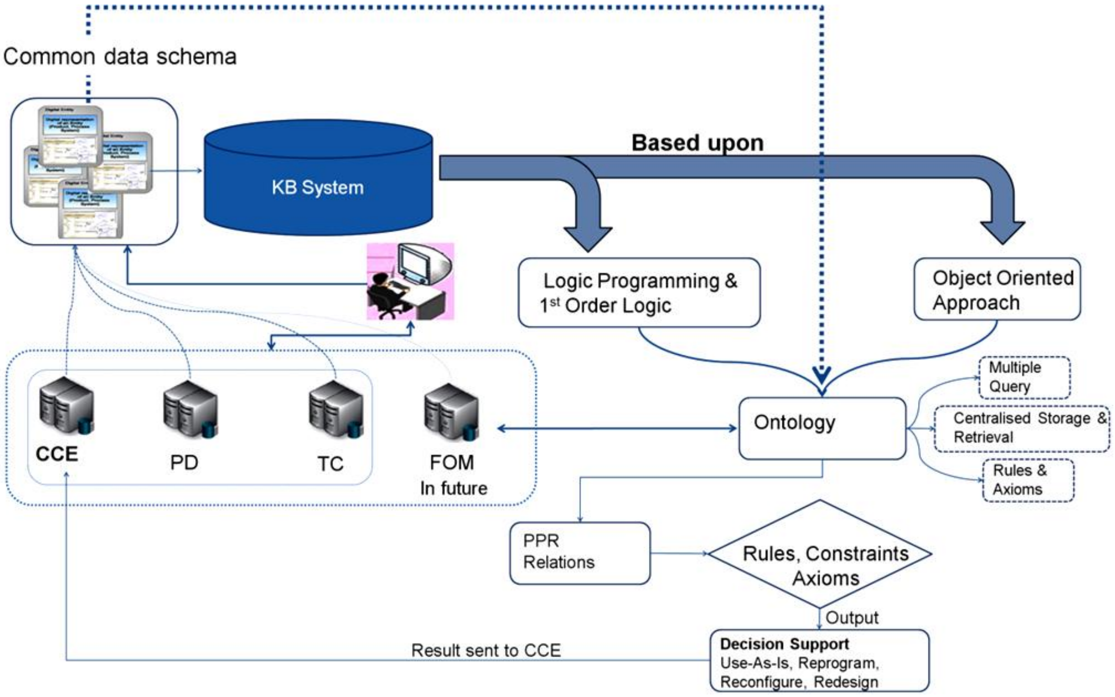


Figure 29: ALDIMS knowledge based system generic architecture

Figure 29 shows a schematic of the system specifications and internal structure. The data from the existing applications is extracted and converted to a common data structure in the form of ontologies. This is based upon logic programming, first-order logic and object-oriented approach. The information from separate applications is brought into the ontology as a common data structure and is refined in the form of rules and axioms for the PPR entities to provide decision support for the engineers. The current system inputs and outputs are summarised below:

4.4.1. Inputs and Outputs of the KB system

The KB system provides an infrastructure to formally document the results of the simulation analyses and decisions made. The following are the inputs and outputs for the KB system:

4.4.1.1 KB Inputs

The following applications currently provide direct inputs for the system: CCE, TC and PD. In addition to this, expert opinion of the experienced engineers can be transformed into explicit knowledge as a direct input into the system.

There are many applications and parallel activities which are being carried out during the assembly line design and reconfiguration process, the outcome of those activities is populates the TC and PD databases.

Digital Pre-Assembly (DPA): From the program start point to its completion point, CAD data is being continuously developed, analysed, modified and published. This continuous process to develop and validate product, manufacturing bill of process, tooling standards, and facilities compatibility is called DPA.

Virtual Build Event (VBE): There are different publication points during the DPA process where the finalised data is published. At each publication point, a discrete event involving cross-functional teams to review the product and manufacturing process is scheduled, which is called VBE. The VBE is aimed to resolve potentially problematic issues for those product parts and assembly stations which are considered likely to create assembly problems, ergonomic issues etc.

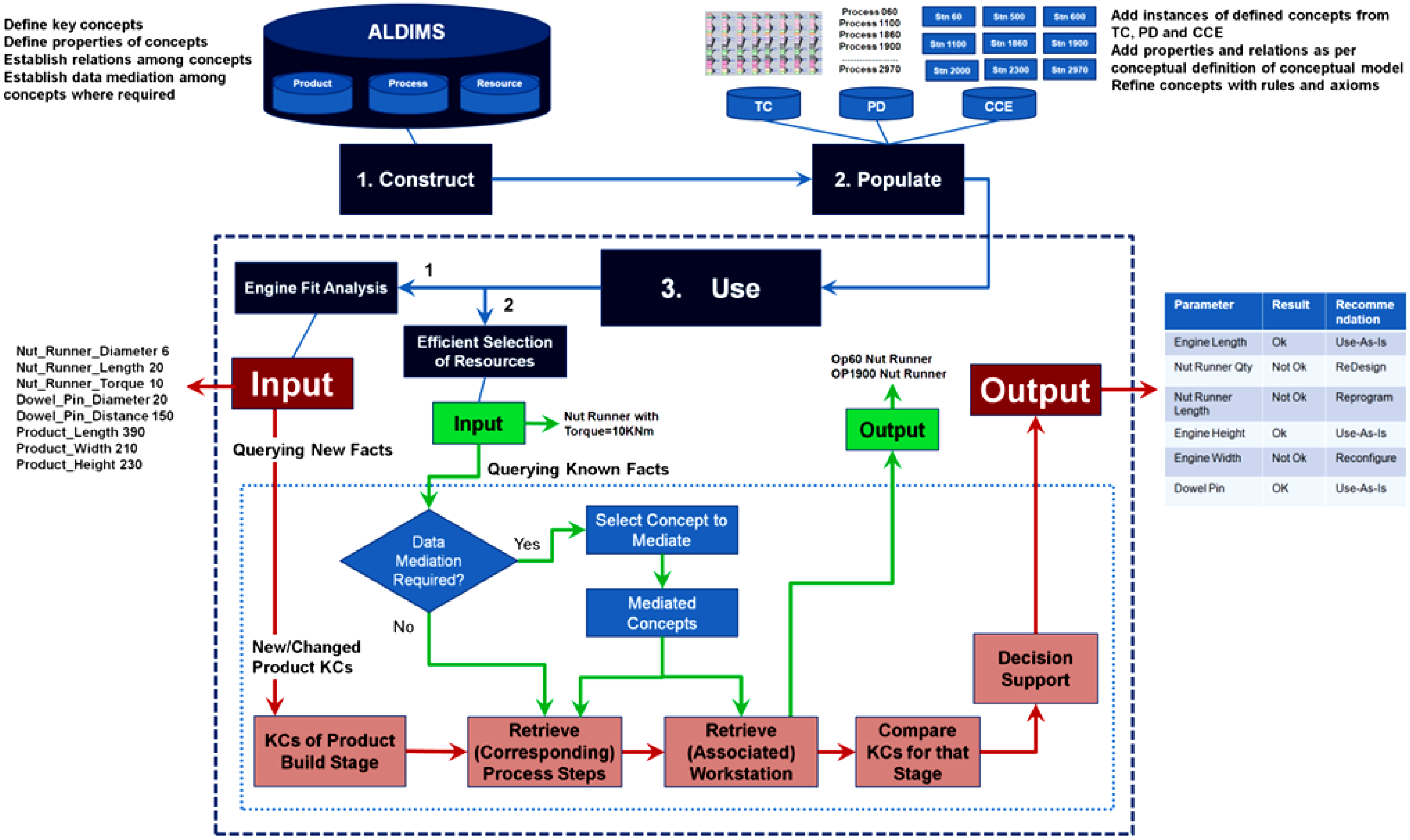
Slow Build Event (SBE): In an SBE, a mock-up study of a manual or semi-automatic workstation is carried out in detail by engineers looking at the physical parts and considering the associated processes.

Installation and Commissioning: The lessons learnt during these phases may be recorded in the KB system.

4.4.1.2 KB Outputs

As described before, the KB system output may be queried for existing facts as well as for the new facts. Therefore it can be used for the efficient retrieval of structured information as well as for consultation on changed parameters of a product and/or process and their effects on the existing machines (rapid constraint evaluation and reconfiguration management).

Figure 30 summarises the construction, population and usage scenarios and recaps on the interaction mechanisms with inputs and outputs.



Define key concepts
Define properties of concepts
Establish relations among concepts
Establish data mediation among concepts where required

Add instances of defined concepts from TC, PD and CCE
Add properties and relations as per conceptual definition of conceptual model
Refine concepts with rules and axioms

Nut_Runner_Diameter 6
Nut_Runner_Length 20
Nut_Runner_Torque 10
Dowel_Pin_Diameter 20
Dowel_Pin_Distance 150
Product_Length 390
Product_Width 210
Product_Height 230

Parameter	Result	Recommendation
Engine Length	Ok	Use-As-Is
Nut Runner Qty	Not Ok	ReDesign
Nut Runner Length	Not Ok	Reprogram
Engine Height	Ok	Use-As-Is
Engine Width	Not Ok	Reconfigure
Dowel Pin	OK	Use-As-Is

Figure 30: Summary of case study and support activities usage

The ALDIMS tool is first constructed on conceptual level by introducing key concepts i.e. product, process and resource, their properties and relations among them. The system is then populated with instances of the concepts. The populated system is ready to be used for assembly line design / reconfiguration activity, the two main target uses are (i) engine fit analysis and (ii) resource library characteristics for efficient querying.

4.5 Comparison 'As-Is' Vs. 'To-Be' Approach

It is intended that a comparison of the existing practices among engineers against the proposed future state will outline the benefits of using CCE modular tool with KB system and provide a roadmap to increased efficiency and streamlined information exchange. To better understand the potential advantages of the ontological knowledge base system implementation at the Ford motor company, a comparative analysis of one of the same tasks, design / reconfiguration of engine assembly line, done before and after the implementation of the ontological knowledge based is given below:

4.5.1. Shift in Machine Design Activity

The Figure 138 shows a very high level conceptual interpretation of the complex process of designing or reconfiguring a new engine assembly line at automotive OEMs e.g. Ford. The BDA project has helped Ford company to take control of the machine design activity while the suppliers should be responsible for machine build only. Currently machine design and build is almost solely in the hands of supply partners. A simplified, high-level responsibilities and interactions are depicted in Figure 31.

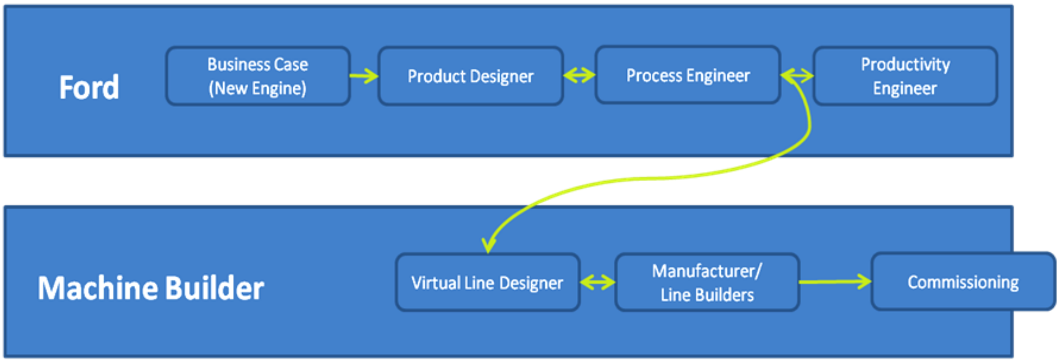


Figure 31: Current business process sequence in condensed form

The business case for design / reconfiguration of engine assembly lines within automotive sector, taking into account OEM and supplier view is considered. After the introduction of the CCE tool and the ALDIMS, the prevalent activities for the assembly line design / reconfiguration are modified as shown in Figure 32.

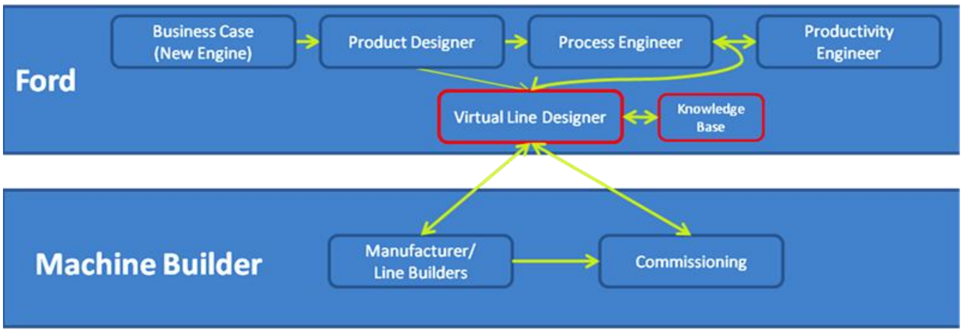


Figure 32: Direct impact of KB on the machine building activity

The task of assembly line design / reconfiguration is broken down into simple steps (reducing complexity) and compared with the existing practices and potential improvements in the activities is presented below:

Task: Process engineer needs to commission a new powertrain assembly line

Current activities in condensed form:

- Get the new / changed product (engine) 3D CAD model from Teamcenter.
- Evaluate changes in the product
- Define new BoP through experience and simulations / VBE / SBE / DPA activities.
- Pass on the data to machine tool builders with suggestions
- Machine builders design/build assembly machines
- Ford engineers physically test and approve machines at vendor's site
- Commissioning of machines at Ford plant
- Re-test commissioned machines for potential hurdles / obstacles with product, plant or other machines etc.
- Focus on specific areas of improvement and apply changes to machines/line
- Re-test and finalise.

With the introduction of the ontological KB system, the same task is reduced and facilitated in the following way, future activities in condensed form:

- Get the new / changed product from Teamcenter, retrieve the KCs of the changed product and use the KB system to automatically evaluate the effect of product changes onto the existing machines and find appropriate components for the changed machine / line and/or design new ones with the help of defined rules.

- Virtually build new or reconfigure existing machines at Ford with CCE tool (along with KBS decision support) using pre-built machines, modules and components with the available option of a complete relational BoM, BoP and BoR in the KB system.
- ‘Virtually’ test and commission the affected machines and approve the final build virtually at Ford in collaboration with machine tool builders and make changes virtually at Ford, if required.
- Machine builders build the machines as per virtual specifications approved by Ford and commission the machines at Ford plant on time and with much more confidence.

The comparison shows that there will be a fair amount of automation, therefore, reduction in lead-time, in the design / reconfiguration of the engine assembly line process. This is one of the possible business processes where ontological KB system can help improve the efficiency and save costs.

4.6 Conclusion

This chapter has provided a detailed explanation of the conceptual and working model of the research and ontology building process for the PPR facets. In addition to this, the overall structure of the ontological KB system, user interface characteristics as well as the underlying technique for automatic evaluation of the product change effects is explained. The formulation of rules and axioms and generic guidelines for the construction of rules is presented. Finally a simplified example of the rule format is described to explain the research concept and advantages of knowledge based support. The support of a KB system in quick analysis of product change effects is desirable because human judgement is prone to errors as well as may overlook critical areas, especially when the change has an indirect effect on an assembly flow line.

CHAPTER 5. ALDIMS: STRUCTURE, ARCHITECTURE AND IMPLEMENTATION

5.1 Introduction

This chapter introduces the implementation and execution platform of the proposed research concept in terms of technicalities and mechanics of the ontological KB system. The main objective of the chapter is to introduce the technical foundations upon which the prototype system is built upon as well as the context application situations and characteristics of the architecture.

The implementation of the developed prototype system is part of an ongoing research project called BDA in collaboration with Ford Motor Company, UK. The aim of the BDA project is to introduce modular, reconfigurable automations systems and the author's research complements the concept by providing an integrated platform for cross-functional knowledge support to store, structure and reuse processed information.

The next sections provide an insight into the system design process and implementation framework from the technical aspects of the used methodologies

5.2 Structural Analysis of ALDIMS

A knowledge base has four main functions[59], the first to act as a foundation for information management; the second function is to provide inference mechanism to generate more knowledge from available information, the third function is to provide a standardised interface among several related applications and the fourth function is easy access for global clients. Therefore, a KB system should be able to transform enormous collections of superfluous data to refined and processed information containing semantics and reasoning capabilities. These requirements have been incorporated in the author's system. The knowledge based middleware consists of three prime notions (i) messaging i.e. a transportation mechanism from source application to the target application or user interactions (ii) mediation i.e. transformation of information in the required

[59] Hopgood (2000) has described three functions of a knowledge based system while the author has added the fourth one in the current global environment.

context / format and (iii) workflow i.e. how information is moved between and across organisational boundaries in relation to business requirements. The ontology based knowledge oriented systems facilitate the user to query and process information based upon rules and axioms from distributed heterogeneous systems.

5.2.1. Layered Architecture

The conceptual architecture of the ontological knowledge based system consists of three 'extensible layers' as shown in Figure 33. An extensible architecture is essential so that the system evolves over time and support changes in the data structures and/or DBMSs. A layered architecture maintains simplicity and modular approach as shown and explained below:

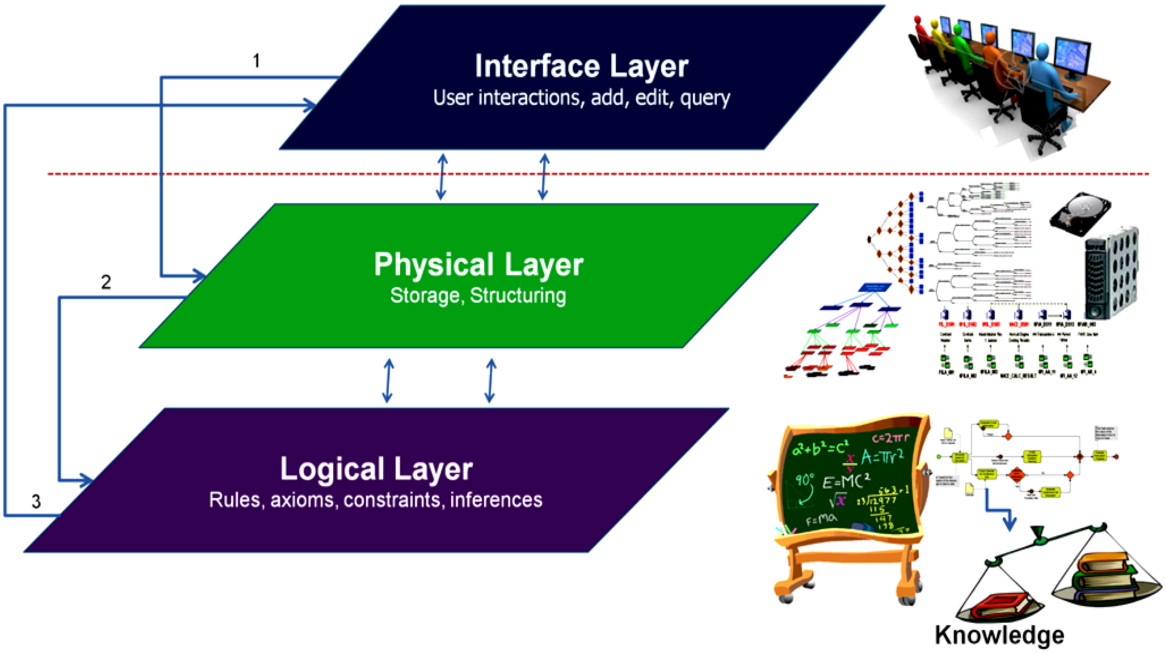


Figure 33: Multi-layered architecture of the KB system

The three layers are: (1) User interface layer (front end); (2) physical layer (knowledge modeling layer) and the (3) logical layer (intelligent decision using rules and axioms layer). The three layers act as architectural modules and simplify the development process. The use of ontology also categorically distinguishes between conceptual knowledge and factual knowledge of the domain. The ontology access layer is provided by a user friendly form (JSP[60] based form) which is accessible through internet. The user form contains options to add, edit and query the knowledge based system in an easy and straightforward manner without going into technicalities of the system. The information in the knowledge base is stored in a semantically enriched text file which can be converted to OWL, XML or RDF formats, if required. The ontology provides a meta-model while the ontology language specifies the structure and semantics of the meta-model. The access to the KB system can also be made through direct access to the IDE[61] ontology editor i.e. WSMT editor. The physical layer contains the format specification, schema and storage layer. The physical layer may be further decomposed into two sub-layers i.e. (i) model / abstract layer and (ii) information layer. The abstract layer constitutes a meta-model of the domain of interest which defines concepts, attributes and relations, while the information layer contains all the necessary information at the instance level in the structured form as defined in the abstract layer. Finally, the logical layer contains all the rules and inference mechanisms which are asserted to concepts, attributes and relations to express constraints, extensions, refine semantics and to add arbitrary knowledge to the defined concepts.

5.3 Implementation / Execution Platform

As described before, the prototyped KB system is developed in WSML ontology. The Integrated Development Environment (IDE) is served by a WSMT[62] editor which is an object-oriented ontology shell and already contains a reasoning engine i.e. Integrated Rule Inference System (IRIS[63]) which serves as an extensible reasoning framework for WSML-Core and WSML-Flight variants of the WSML[64]. The knowledge represented in ALDIMS is retained and extended with the help of classes, attributes of classes, relations among classes and rules and methods of different types.

To run the ALDIMS tool, the Windows / Linux operating system should be installed on the system, 77MB of memory and 120 MB of disk space should be available. The technical implementation is a combination of the following technologies: i.e. a WSMT editor based on WSML, IRIS Reasoner, Geronimo Server and JSP (Jena) framework. The inter-relationship of

[60] Java Server Page

[61] Integrated Development Environment

[62] http://sourceforge.net/projects/wsmt/ , The WSMT shell provides easy connectivity with legacy DBMS through eclipse based APIs.

[63] IRIS – An open source project: http://sourceforge.net/projects/iris-reasoner/

[64] http://tools.deri.org/wsml2reasoner/

these entities in the developed system in terms of the deployment and execution specifications is shown in Figure 34. The ALDIMS tool can be accessed through a web page as well as directly through the WSMT editor which essentially is the front end of the application. The WSMT editor is the main technology component while the IRIS reasoner is the infrastructure component[65] of the application. The rest of the infrastructure components supporting the execution of the application are at the backend. The configuration files i.e. query and property files, provided in the setup to make the application flexible, form the deployment components[66] of the application.

[65] The infrastructure component provides additional support to the main technology component of the application

[66] The deployment component ensures proper deployment of the software application

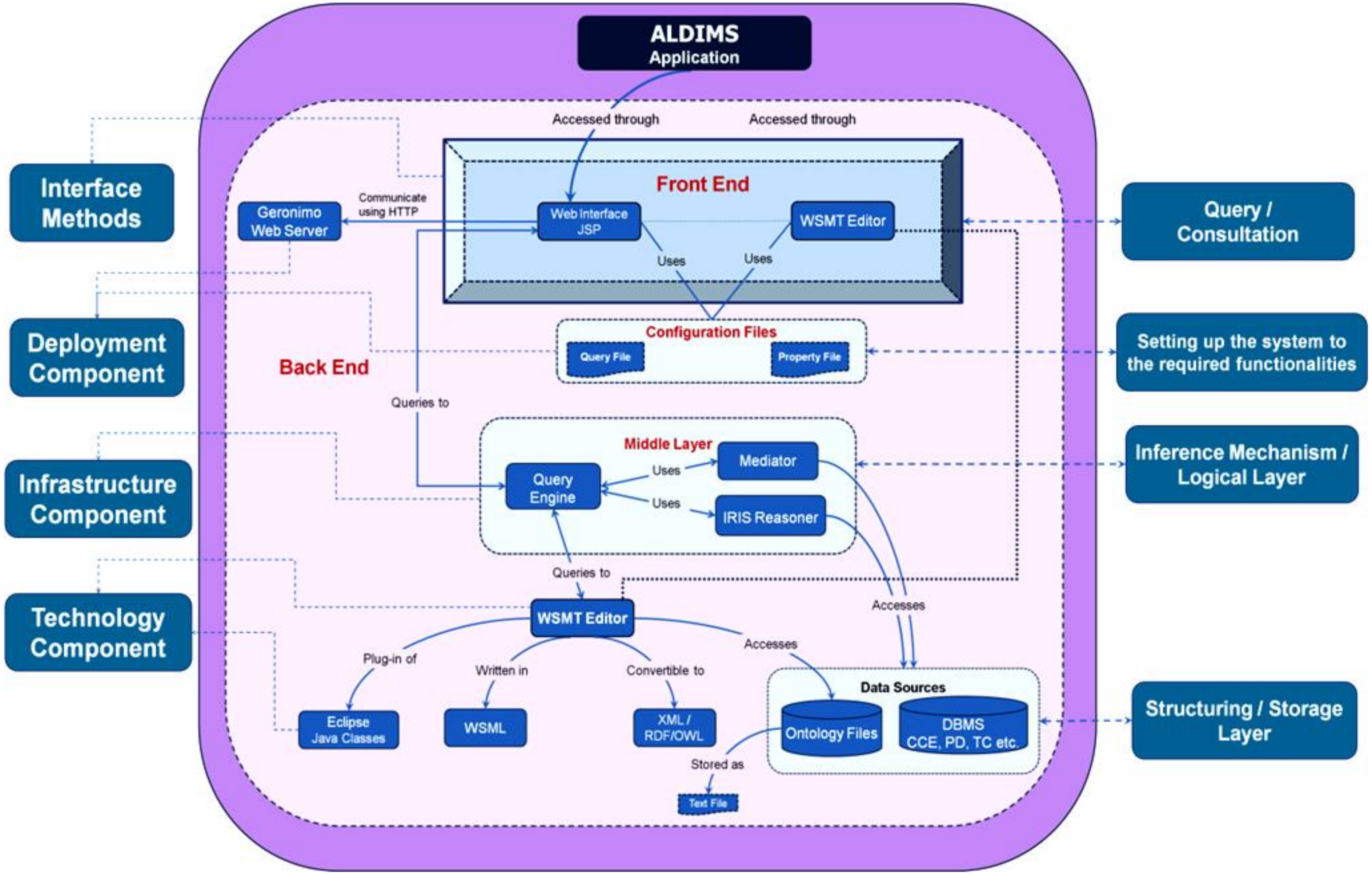


Figure 34: The configuration of the platform for the ALDIMS execution

The Tomcat Geronimo server[67] is used to run web pages and imitate a real scenario where it will be replaced with the server of the Ford Company. The JSP form is the access layer, it runs on Geronimo server and communicates using HTTP, with which it is possible to support hyperlinked web pages in HTML format. HTTP is a document-based, client-server protocol for data transfer on the internet (Gourley and Totty, 2002). Typically a host connects to a server at a particular URL (internet address) and requests a web page which is returned in HTML (Gourley and Totty, 2002).

The WSMT editor is the application layer of the system, the data is stored as object-oriented structure with internet accessibility for search and retrieve of the required knowledge. The information of the three relevant domains i.e. PPR is modelled as ontology, implemented in WSMT and structured as a KB system. There is a provision to import data from other applications into the ontological KB system through DBC-API.

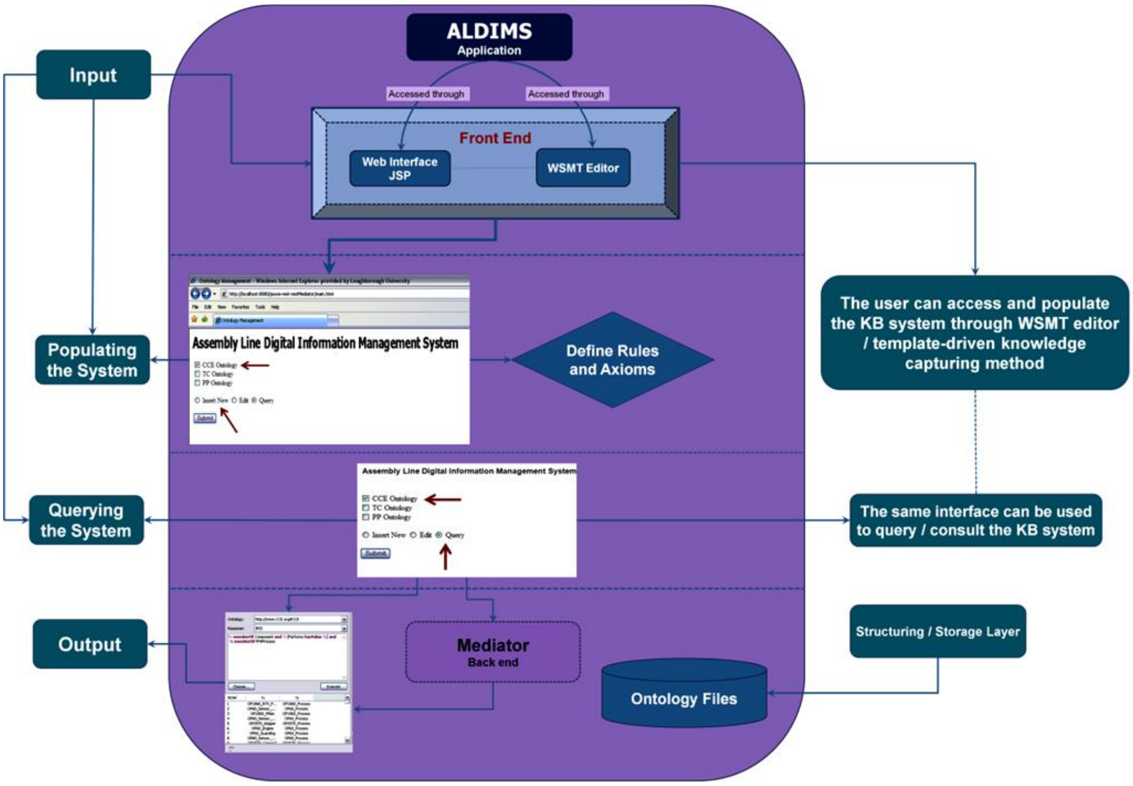

ALDIMS
Application
Input
Accessed through
Accessed through
Front End
Web Interface JSP
WSMT Editor
Assembly Line Digital Information Management System
CCE Ontology
TC Ontology
PP Ontology
Insert New
Edit
Query
Submit
Populating the System
Define Rules and Axioms
The user can access and populate the KB system through WSMT editor / template-driven knowledge capturing method
Querying the System
The same interface can be used to query / consult the KB system
Output
Mediator
Back end
Structuring / Storage Layer
Ontology Files

Figure 35: The ALDIMS application: populating and querying

[67] The system works both with Geronimo and JBoss application servers

The ALDIMS application, once deployed, can be populated with the instance-base i.e. the PPR entities as shown in Figure 35. The application facilitates input through template-based forms. The input forms ensure the application is fed with the correct knowledge model as defined in the ontology. The input provided will be saved as concepts, properties of concepts or relations among concepts of any of the three entities i.e. PPR. Once the concepts are defined, rules and axioms can be made on the defined concepts to further refine the inputs.

5.3.1. Characteristics of the Architecture

There are two ways to access the ALDIMS application, direct access through WSMT editor and the access through a JSP form through the internet. As the system may need to be accessed from geographically distant locations therefore the access is designed to be distributed. The knowledge represented in the ALDIMS is in the form of objects which can be reused. The ALDIMS tool is not a standalone system, rather it is accessible through the web anywhere and at anytime. This enhancement makes the system available on every engineer's desk. The number of users working simultaneously on an activity of this type is usually more than one, this is because the complexity of the problem. Considering this requirement, the ALDIMS is built to be accessed by many users at a time. The application layer protocol i.e. HTTP, HyperText Transport Protocol / HyperText Transfer Protocol[68] is used for communication when accessing the system through a Web browser. The system understands HTTP and can act both as a Web server and Web client.

5.3.2. Sequence Diagrams / Event Diagrams

A sequence diagram[69], is a UML diagram, used to clarify event sequences to reach to the required outcome. These describe how the activities function within the system with one another and in what sequence. Hence the diagrams model the flow of logic through step by step interaction details[70]. As described before, there are two ways by which the KB system can be accessed (i) direct interface and (ii) interface through web based form.

[68] http://www.w3.org/Protocols/

[69] Interactions between entities / objects are drawn as horizontal direction arrows which represent messages sent and replies received in the order in which they occur. Lifelines / boxes are drawn around sets of arrows to show control structures, events or actions. Time is represented in the vertical direction representing the correct order of interactions of the header elements which are displayed as rectangles horizontally at the top of the diagram, time passes from top to bottom, the lowest the latest. Actor interacts with the system, objects interact within the system, separators represent interface or boundary between units of the system. Action is taken by usually an actor, object or unit and the sequence of actions is started.

[70] http://www.agilemodeling.com/artifacts/sequenceDiagram.htm

5.3.2.1 Direct WSMT interface

The sequence diagram for the direct WSMT interface is shown in Figure 36.

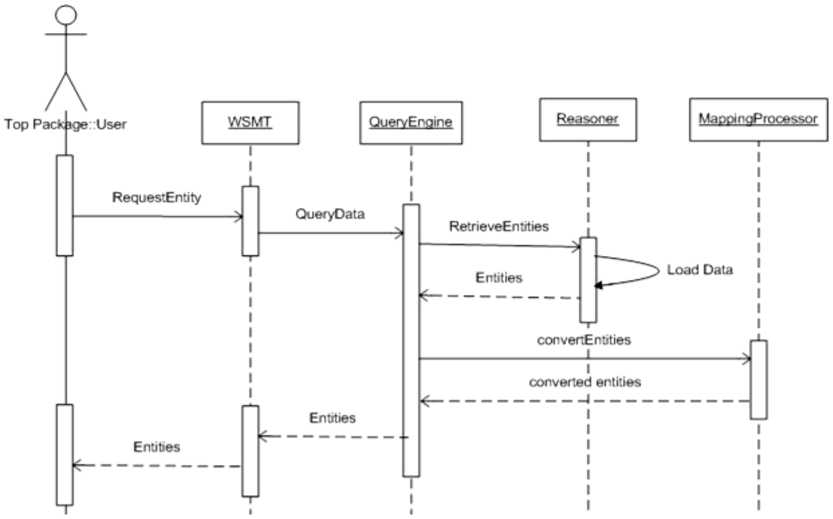


Figure 36: Sequence Diagram for direct WSMT interface

There are four basic objects / lifelines for this scenario i.e. WSMT, Query Engine, Reasoner and Mapping Process. The arrows represent the messages while the dashed lines show the return values. As the user accesses the WSMT editor directly, the interaction process starts. The editor reaches out to the query engine and retrieves entities through reasoner. If the user requests mediated data, the reasoner accesses the mapping files and retrieves mediated data otherwise non-mediated data will be returned to the user as shown in Figure 36.

5.3.2.2 Interface through web page

As described before, the user may access the ALDIMS application through web based form. The only difference with the form access is that instead of WSMT editor, the access is granted through the JSP form with the same process steps happening at the back end as through direct access with the user oblivious of the background processes as shown in Figure 37.

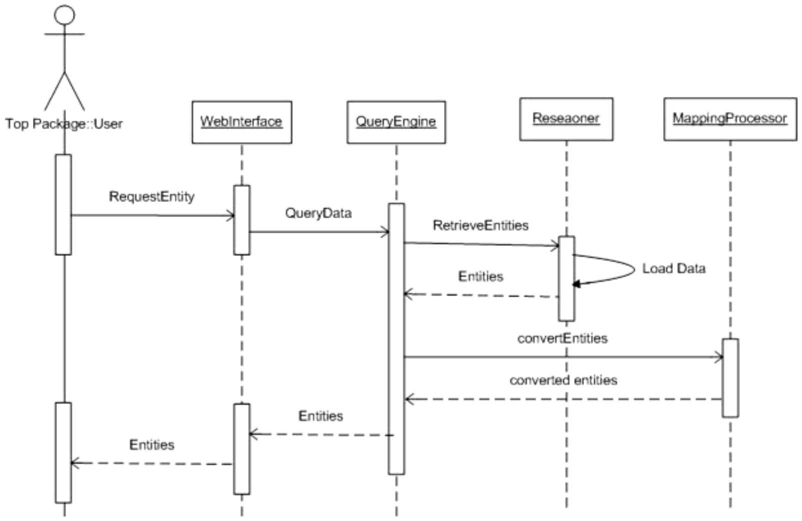


Figure 37: Sequence diagram for web page interface

5.3.3. Reasoning in the KB System

Two types of reasoning methods are provided, the one is the default built in reasoning system and the other reasoning engine is to execute defined axioms.

1. Logical Constraints / First-Order Reasoning
2. Axiomatic Constraints / Rule Based Reasoning

The most intuitive reasoning is provided by the ontology itself. The domain is transformed into classes, classes have properties and relations with other classes in the domain, thus, forming a primary relational and restrictive reasoning. The WSML is based on different logical formalisms, namely, Description Logics, First-Order Logic and Logic Programming. The reasoner ensures the ontology sticks to the conceptual definition declared in the abstract model of the domain. The second reasoning provided by the ontology-based knowledge modeling is based upon a dedicated reasoner called the IRIS reasoning engine. The IRIS engine helps to reason based upon the rules and axioms defined in the ontology.

Ontological instances are the input of the system and the rules operate on ontological instances. The expressiveness of the rules determines the degree to which inferences can be made within the domain area defined in the ontology. Ontology access layer is implemented through a form designed in JSP. The form enables the users to access and query the three main modules of the KB system i.e. insert new, edit and query modules. The knowledge stored in the KB system can be accessed through two user interfaces. One of them is direct ontology editor access and the other one is user form access. For the clients who have got more experience in knowledge based system or an IDE, a direct access to the ontology editor is possible while for the novice users the system may be accessed through user friendly form designed in JSP and accessed through Apache Geronimo server[71]. Both the interface options serve the function to provide direct access to the knowledge stored in the KB system.

The interaction with the KB system starts when the user accesses the system through a client application. The system is a client-server based application and to access the server, internet connection must be available. If the query made in the request access is valid then, it is passed on to the physical layer where the structured information is saved and the required information is pulled out and processed by the logical layer. This may include mediation, rule checking or relational associations etc. the processed information is returned to the user for further analyses. This forms a collaborative work facility with global access to distributed data bases.

5.4 Populating the System

The original plan of the ontological KB system was to introduce automatic retrieval of database objects into the KB system, however, this was not achievable due to the fact that direct access to the Ford's database applications was not readily possible and the temporary server created for file sharing EDMS[72] did not follow a consistent pattern of file structure. This was classified as a low priority work by the end user and currently left as future work plan. At present, the access to the database objects is imported into the KB system manually. Though the import process is manual nevertheless it is template based therefore less prone to errors. A stable ready-to-use trial system with immediate impacts is demonstrated with focus on process not on technology.

The Prototype version of the ALDIMS tool is fed with the required inputs manually as shown in Figure 38.

[71] http://geronimo.apache.org/

[72] Electronic Document Management Software – a temporary file sharing system prevalent at Ford Company

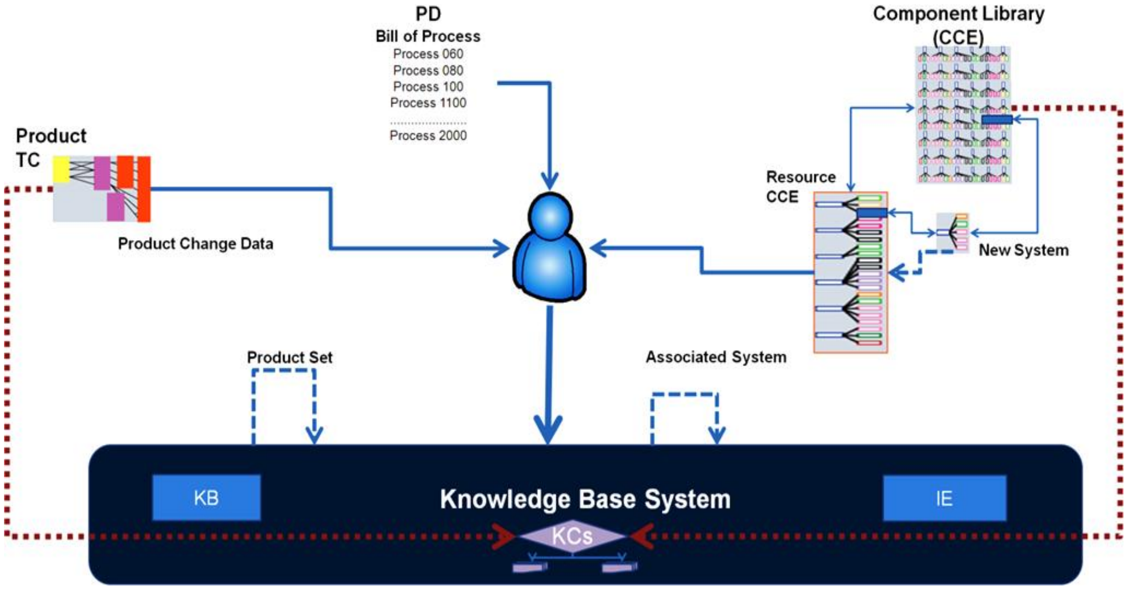


Figure 38: The KB system is populated manually

The interdisciplinary organisation of knowledge is initially being carried out manually, the test dataset was constructed and refined manually and incrementally, mapping among disparate entities established and rules and axioms were verified.

5.5 KB System User Interface

There are many designs and techniques for the user interface of a software system. Iqbal (2009)[73] has defined a few basic characteristics of a successful user interface including: clear, concise, familiar, attractive, efficient etc. while clarity and efficiency are the most important attributes. The interface design intended to be simple and user friendly with the known fact that simplicity shall improve usability. The three major types of user interfaces are (Oliver, 1994):

Command-line interfaces: In this type of interface the input is provided by typing a command string and the output is provided by similar string. The input is given through keyboard and output is received on the computer monitor.

Graphical User Interfaces (GUI): In this type of interface, the graphic capabilities of computer are used to make the interface easier and more user friendly. Usually mouse (sometimes mouse and key board) is used for the input. The output is again provided on monitor in graphical form.

[73] www.usabilitypost.com

Web Based Interface: any interface type which is accessible through web page is a web based interface. With the web based interface, there are many further options to be selected so as to make the interface interactive.

The current user interface is a GUI interface with a combination of "menu" and "form filling" type. This type of interface is provided for easy navigation through the system as well as getting inputs from the user when required. The next section explains the underlying characteristics of the developed system. The current ontological knowledge based system is provided with 03 modules i.e. (i) the search module, (ii) the knowledge generation module and (iii) the edit module.

5.6 ALDIMS User Interface Characteristics

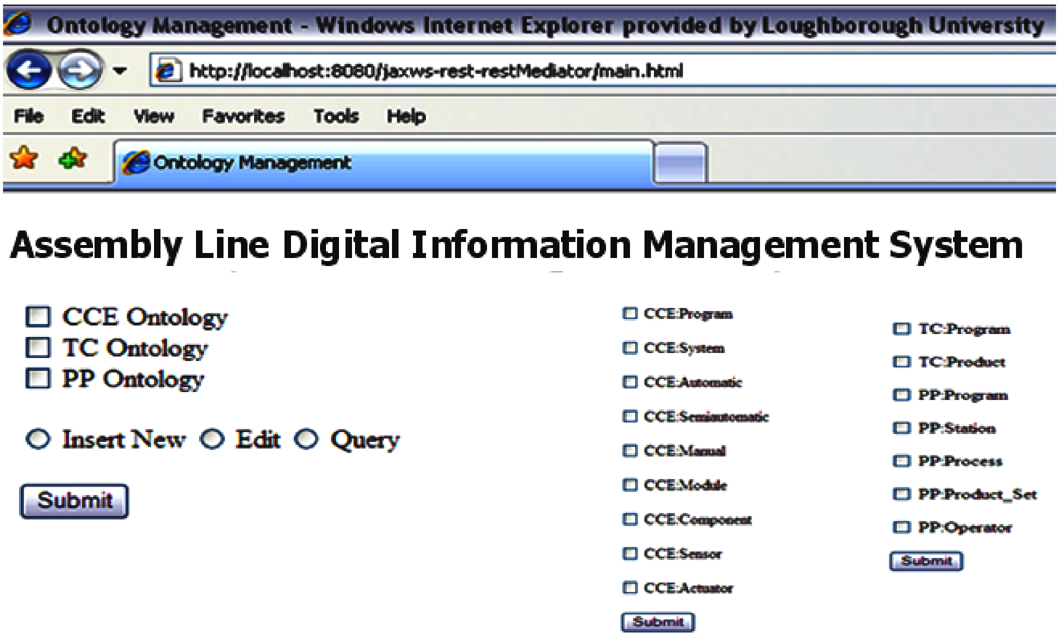
The ALDIMS application can be accessed through two interaction mechanisms:

1. Access through Web Browser
2. Direct Interaction

5.6.1. Web-Based Interface

These are an extension of GUIs. Input and output is provided by generating web pages which are transported via the web and users can view through a web browser. Internet or a Local Area Network (LAN) and viewed by the user using a web browser program. This enables to use graphical properties of computer as well as distributed nature of web both at the same time.

Considering the future usage of the developed system, web-based graphical user interface has been chosen for easier and multi-use of the system. With the web based interface many users situated at geographically distant locations can access and interact with the system. A simple interface is shown below in Figure 39.


Ontology Management - Windows Internet Explorer provided by Loughborough University
http://localhost:8080/jaxws-rest-restMediator/main.html
File Edit View Favorites Tools Help
Ontology Management

Assembly Line Digital Information Management System

□ CCE Ontology
□ TC Ontology
□ PP Ontology

○ Insert New ○ Edit ○ Query

Submit

□ CCE:Program
□ CCE:System
□ CCE:Automatic
□ CCE:Semiautomatic
□ CCE:Manual
□ CCE:Module
□ CCE:Component
□ CCE:Sensor
□ CCE:Actuator
Submit

□ TC:Program
□ TC:Product
□ PP:Program
□ PP:Station
□ PP:Process
□ PP:Product_Set
□ PP:Operator
Submit

Figure 39: Graphical user interface for ALDIMS through web browser

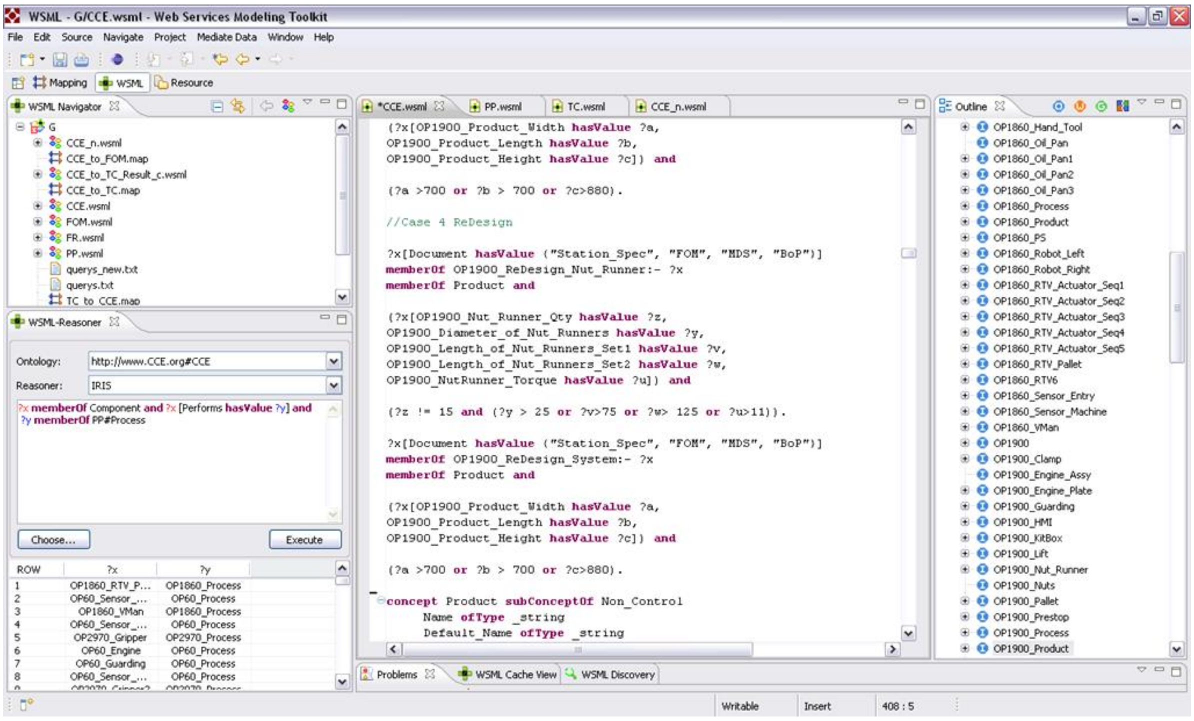
Interfaces should be designed to reduce the complexity of a given framework. Information should be presented and organised so that only relevant information is passed on to the user in a simple manner. The same is addressed while designing the interface of the system.

5.6.2. Direct WSMT Interface

The WSMT editor is an IDE based on the WSML language. The WSMT IDE editor is similar to Eclipse Java IDE in many respects. A detailed description is provided in the form of a tutorial on the WSML project websites[74], describing complete guidance from installation to creating first ontology, however, a brief introduction is provided here. WSMT editor is a simple and fully featured ontology development IDE. The WSMT version 1.4 is used for development of ontologies in the current research which can be downloaded free of charge from the internet[75]. The WSMT requires Java 5.0 or higher installed as the default JRE on the system. One of the features of the editor is a simple GUI with self explanatory instructions. It has 06 major sections, discussed briefly in the next sections.

Figure 40 provides a snapshot of the WSMT editor interface of the developed system.

[74] http://www.wsmo.org/TR/d9/d9.1/v0.2/20050425/ ; http://www.wsmo.org/TR/d9/d9.2/v0.1/20050321/
[75] http://sourceforge.net/news/?group_id=154080



Ontology: http://www.CCE.org#CCE

Reasoner: IRIS

```
?x memberOf Component and ?x [Performs hasValue ?y] and
 ?y memberOf PP#Process
```

ROW	?x	?y
1	OP1860_RTV_P...	OP1860_Process
2	OP60_Sensor_...	OP60_Process
3	OP1860_VMan	OP1860_Process
4	OP60_Sensor_...	OP60_Process
5	OP2970_Gripper	OP2970_Process
6	OP60_Engine	OP60_Process
7	OP60_Guarding	OP60_Process
8	OP60_Sensor_...	OP60_Process

```
(?x[OP1900_Product_Width hasValue ?a,
OP1900_Product_Length hasValue ?b,
OP1900_Product_Height hasValue ?c]) and

(?a >700 or ?b > 700 or ?c>880).

//Case 4 ReDesign

?x[Document hasValue ("Station_Spec", "FOM", "MDS", "BoP")]
memberOf OP1900_ReDesign_Nut_Runner:- ?x
memberOf Product and

(?x[OP1900_Nut_Runner_Qty hasValue ?z,
OP1900_Diameter_of_Nut_Runners hasValue ?y,
OP1900_Length_of_Nut_Runners_Set1 hasValue ?v,
OP1900_Length_of_Nut_Runners_Set2 hasValue ?w,
OP1900_NutRunner_Torque hasValue ?u]) and

(?z != 15 and (?y > 25 or ?v>75 or ?w> 125 or ?u>11)).

?x[Document hasValue ("Station_Spec", "FOM", "MDS", "BoP")]
memberOf OP1900_ReDesign_System:- ?x
memberOf Product and

(?x[OP1900_Product_Width hasValue ?a,
OP1900_Product_Length hasValue ?b,
OP1900_Product_Height hasValue ?c]) and

(?a >700 or ?b > 700 or ?c>880).

concept Product subConceptOf Non_Control
     Name ofType _string
     Default_Name ofType _string
```

Figure 40: Accessing the KB system directly through WSMT editor

Complete detail of the interface options of the developed KB system is provided in Appendix 1.

5.7 Summary

The functional aspects and technical architecture of the developed KB system have been elaborated in this chapter. This includes the implementation platform and characteristics of the architecture duly supported by sequence diagrams for the two access methods to the system i.e. direct access and access through JSP form. This is followed by a description of the reasoning techniques built into the WSML ontology which in turn, become part of the ALDIMS. The reasoning helps to deduce inferences automatically based upon known facts.

CHAPTER 6. CASE STUDY SCENARIOS

6.1 Introduction

The chapter provides detail of industrial scenarios where the developed application will be used. The main focus of the research is the realisation of rapid engine fit analysis through parameterised analysis of constraints, as well as the introduction of the most relevant support activities required to achieve the engine fit analysis. This includes initial proof of the concept on the Festo rig, an experimental test case set up on the rig and improving the real case study through lessons learnt on the Festo rig. The pilot implementation at Ford is demonstrated on four diverse workstations on the Tiger assembly line at Ford, and finally the advantages of the support activities are highlighted and conclusions drawn.

6.2 Aim and Objectives of the Case Study

In line with the aim and objectives of the overall research, the case study is designed and conducted to implement and evaluate the suggested research concept through application of the KB system.

The main aim of the case study is to evaluate whether a relational knowledge based system capable of assimilating PPR information can present an integrated view of the domains which would help efficient decision support, effective reuse of program knowledge and rapid digital prototype simulation and in turn more effective final configuration of the engine assembly lines in automotive sector. A brief of the objectives of the case study are described below:

1. To finalise the prototype system design, development, testing and validation to:

   - authenticate the axioms and rules defined in the ontological KB system
   - demonstrate the data mediation concept, its application and potential benefits in the automotive sector by harmonising the localised data from heterogeneous applications.
   - confirm the use and benefit of ontologies to help reuse of existing knowledge in the context of designing / re-configuration of the modular engine assembly line to build new/changed product,
   - verify the concept of efficient search and retrieval of the best component/module of the machine against stated product requirements with the help of semantically enriched information in ontological KB system.

2. To evaluate the viability of automating constraint evaluation can be fairly automated with a correctly modelled domain with rules and axioms defined among the interdisciplinary domain entities i.e. PPR in the automotive sector.
3. To enable a process engineer to quickly check and verify the changed assembly line parametrically.

All the case study scenarios relate to two major business requirements i.e.

- A new production facility is being established, or
- Changes in existing production facility are required.

For scenario one, new machines have to be designed from generic machines while for scenario two, existing machines have to be modified to produce a different variant of an engine. At Ford, development work is in progress to generate virtual models of generic machines. A number of generic machines have already been designed and being evaluated, however, a complete set of workstations constituting a virtual assembly line is not available as yet. Similarly only a few virtual models of production machines currently exist. However it is expected that in near future Ford will get all the necessary 3D models of the assembly line from its vendors.

The case study can be split into two main phases of (i) initial lab-based testing on a small scale rig (e.g. Festo rig) and then (ii) pilot proof-of-concept tests using example engine assembly stations.

The two main business cases are (i) a new production facility is being installed and (ii) reconfiguration in the existing facility is being carried out. One of the objectives of the current research work under the BDA project is the use and implementation of the research results in the automotive industry, see Figure 41.

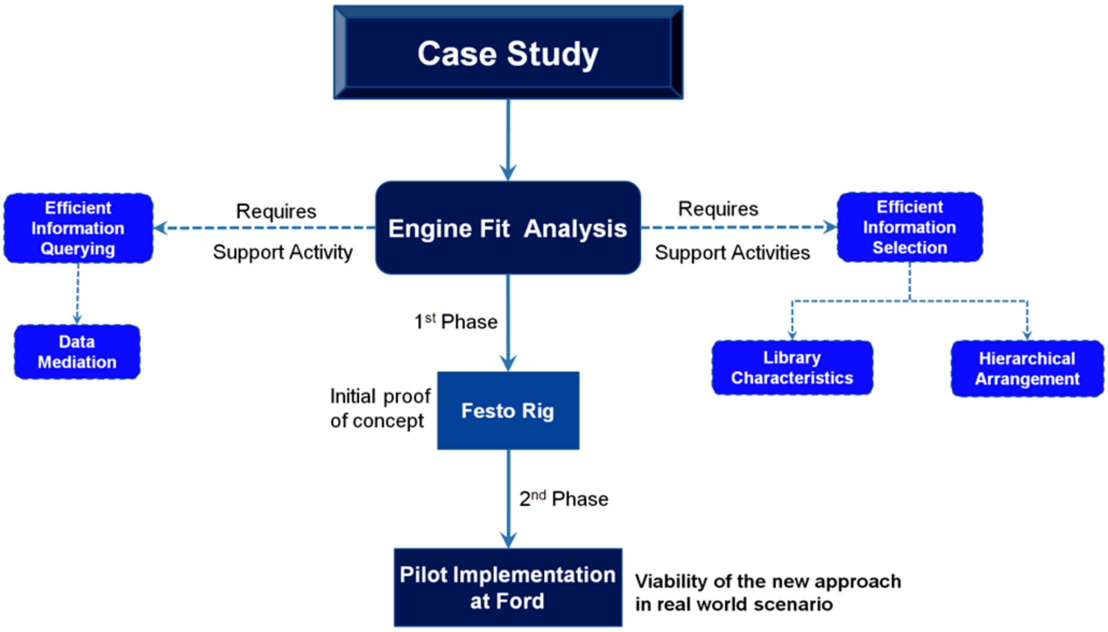


Figure 41: Summary of case study scenarios, industrial situation and proof phase

The case study is focussed on the ‘engine fit analysis’, however, there are a few support activities which are necessary to incorporate into the KB system to make the system robust and viable. The three additional facets are studied and incorporated in the developed system so that a more smooth realisation of the engine fit analysis can be carried out.

6.3 Initial Proof of Concept

In order to prove the concept of ontological based knowledge support for an engine assembly line, it was planned to show the potential benefits on one of the mini training rig installed at the Manufacturing Systems Integration (MSI) lab of Loughborough University. This training rig, called the “Festo Didactic Test Rig”, mimics the large scale real assembly automation systems at Ford. It has got exactly the same interfaces and communication ports (PLCs, IO etc.) as are prevalent at Ford. The purpose of the initial testing on the Festo rig was to verify the robustness of the ontology being used for building rules and axioms.

6.3.1. Festo Rig

This section describes a brief of the Festo rig and the requirements for the demonstration of the research concept. The Festo rig is shown in the Figure 42.

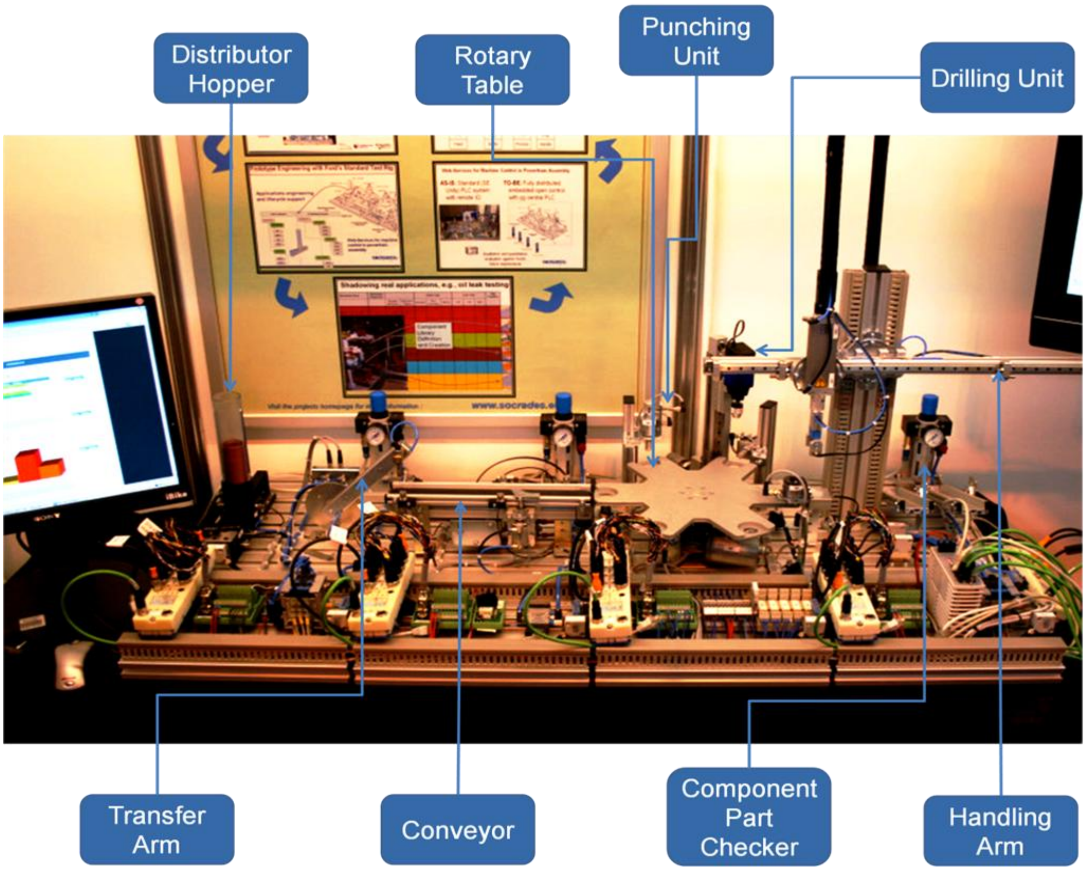


Figure 42: Festo Didactic Test Rig – Ford assembly line mimic

The rig has handling stations (resource), an aluminium workpiece for the stations to work upon (product), sequence and process steps for the stations to follow (process / BoP). The rig has been split down into components, the components are items such as individual sensors and actuators that reside on a single component. Each component been described in terms of its functionality and its input and output requirements.

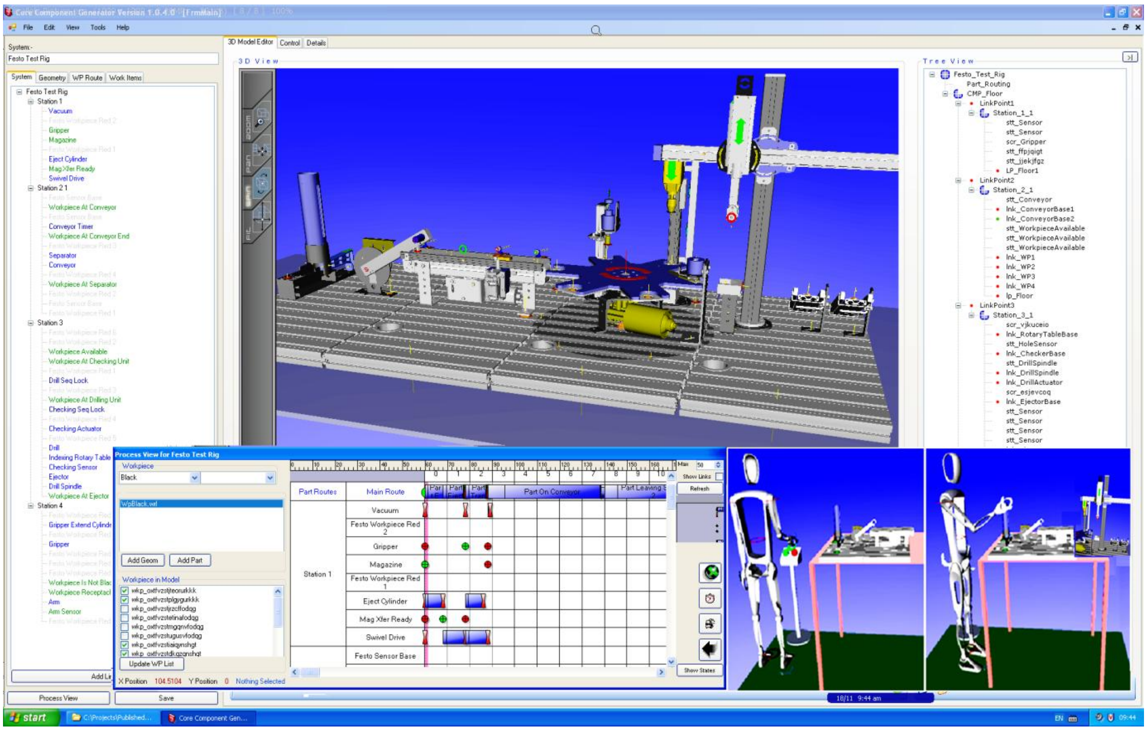


Figure 43: Festo Rig Model in the CCE Tool – VRML Interface

The 3D model of the Festo rig is shown in Figure 43. The model, on one hand, simplifies the rig operations while, on the other hand, it provides options to simulate possible future alterations in the product / resource without actually making any modification in real terms.

A brief overview of the Festo rig operations is depicted in Figure 44.

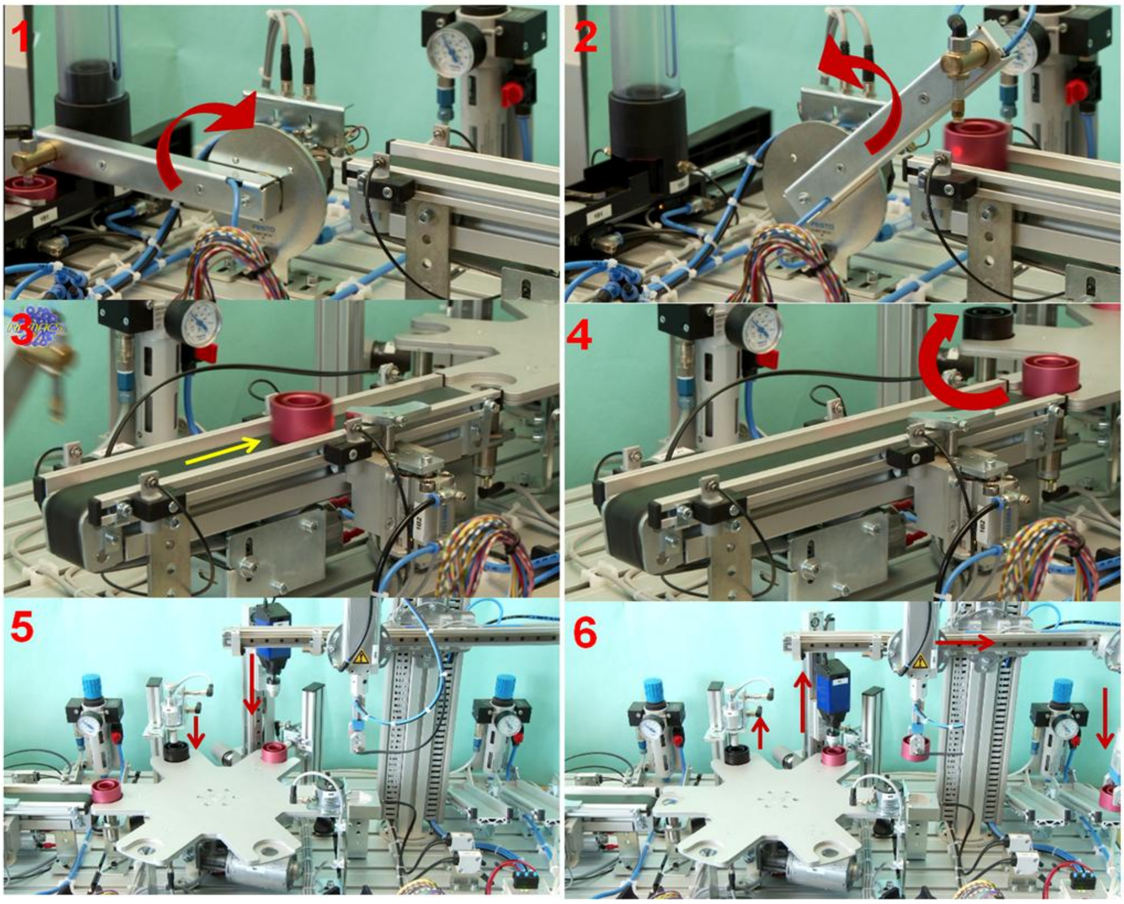


Figure 44: Festo Rig Operations

6.3.2. Festo Rig in terms of PPR Domain Description

The Festo rig is decomposed into modules/components, with each module independent to each other. A description of each module in terms of its components, operational positions and I/O requirements is briefly presented. Table 13 shows the Festo rig in terms of PPR domains.

Product	**Process**	**Resource**		
		System / Workstation	**Components**	
Round Al. Piece	Place the product into the distribution hopper	Operator	Qty	Sensor / Actuator
Round Al. Piece	Workpiece Feed	Distribution Hopper	3	1 Actuator, 2 Sensors
"	Pick & Place	Transfer Arm	3	2 Actuators, 1 Sensor
"	Product flow	Conveyor	5	2 Actuators, 3 Sensors
"	Decide if the hole is drilled ok	Component Part Checker Probe	1	Actuator
"	Moves product to	Rotary Table	5	2 Actuators , 3 Sensors

	indexed positions			
	Punch	Punching Unit	3	3 Actuators
“	Drills the hole	Drilling Unit	3	3 Actuators
“	Moves gripped to set position	Handling Arm	5	3 Actuators, 2 Sensors
“	Main Control	Operator Panel		

Table 13: Summary of Festo test rig described as PPR domains

There are nine resource units including operator with as many process steps, the eight resource units have 25 smaller components in total and one product which travels through the rig. The product (workpiece) on the rig is a circular piece of material aluminum which is either red or black as shown in Figure 45 which is transported, punched, drilled, gauged and sorted. A complete detail of the ontology building process for the Festo rig is provided in Appendix 2.

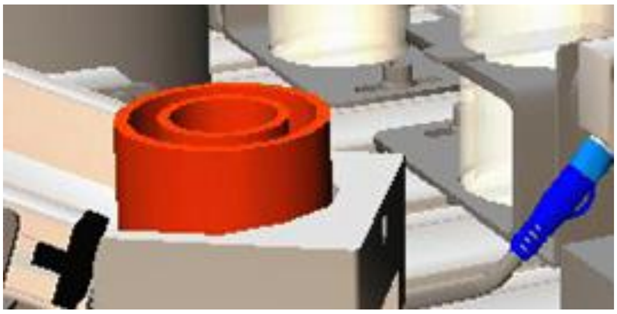


Figure 45: Workpiece on the Festo rig

6.4 Test Case Set-up on the Festo Rig

For the purpose of this thesis i.e. engine fit to assembly lines, the focus is on the development of an approach to enable parametric evaluation which is often related to dimensional changes of the product. Therefore a relevant scenario was implemented on the festo test rig.

The diameter of the workpiece cannot be increased due to constraints at almost all of the stations on the rig, however the height may be increased and an evaluation of the potential effects on the stations is then performed. To increase the height, the two workpieces were stacked together to behave as a single new product. See Figure 46.

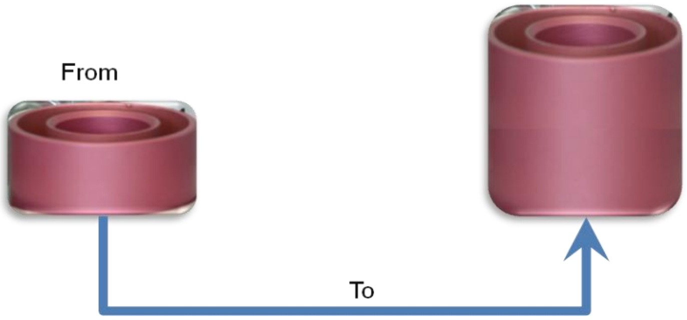


Figure 46: Two workpieces stacked together

The height of the original workpiece is 25mm and the new height is 50mm. The diameter of the workpiece for the test case set up remains unchanged at 40mm.

6.4.1. Run the Changed Product through the Festo Rig

The test case set up for the changed product is ready. Now the product may be run through the entire rig to see effects of the change in product height on different stations. The research main idea is to simulate the PPR system and keep on recording the simulation results in the form of rules and axioms in the KB system.

Four categories of change have been defined, these are (i) use-as-is, (ii) reprogram, (iii) reconfigure and (iv) redesign. These categories provide an insight into the impact of product change on the potential configuration requirement of the existing stations.

The very first station at the rig is distribution hopper, the station keeps the product in the stack form and when ready, the workpiece is pushed to the position where transfer arm can pick up the workpiece. The workpiece is slid down through the opening provided at the base of the hopper. The opening height is 28mm as shown in Figure 47, therefore, the changed workpiece with increased height cannot pass through the opening. Hence it needs to be redesigned for any height greater than 27mm. This rule is included in the ontology. In the current test set up scenario, the product is placed at the transfer arm pick up position manually as shown in Figure 47.

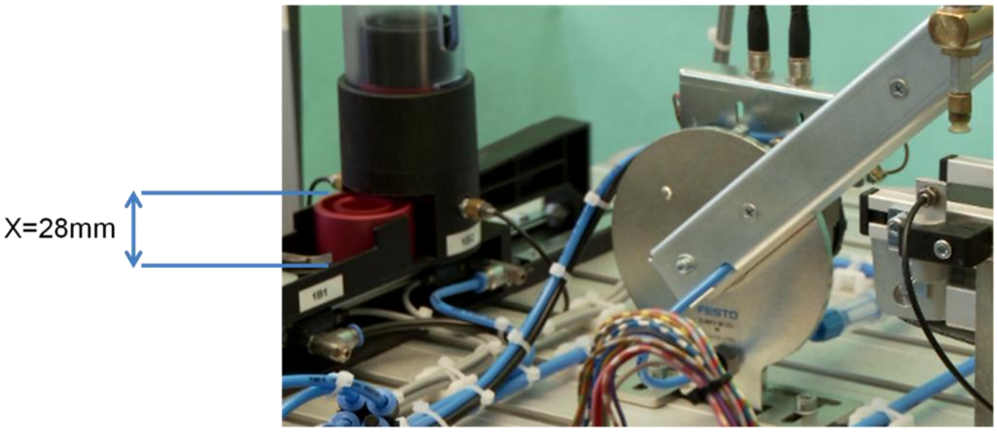


Figure 47: Distribution Hopper

Therefore as the height of workpiece is increased, the distribution hopper ejection slot needs to be increased. In the current test case the workpiece was placed manually at the pickup position rather than changing the distribution hopper.

Now the next step is to transfer the workpiece onto the conveyor. Here again, the transfer arm could not pick up the workpiece. Initially it was concluded that the weight of the workpiece (32gm) has exceeded to the lifting capacity of the suction gripper on the transfer arm. However it was realised that the weight lifting capability of the gripper is more than this.

Suction force = 1.6N (at -0.75 bar)
Weight of the existing workpiece = 16 grams (approx)
Weight of the changed workpiece = 32 grams (approx)
Weight handling capacity = (1.6 ÷ 10) x 1000 = 160 grams

It was the position of the lifting arm which needed to be reconfigured. The transfer arm swings approximately 120° to pickup and release the workpiece. This angular displacement needs to be readjusted by reducing approximately 10° on either side.

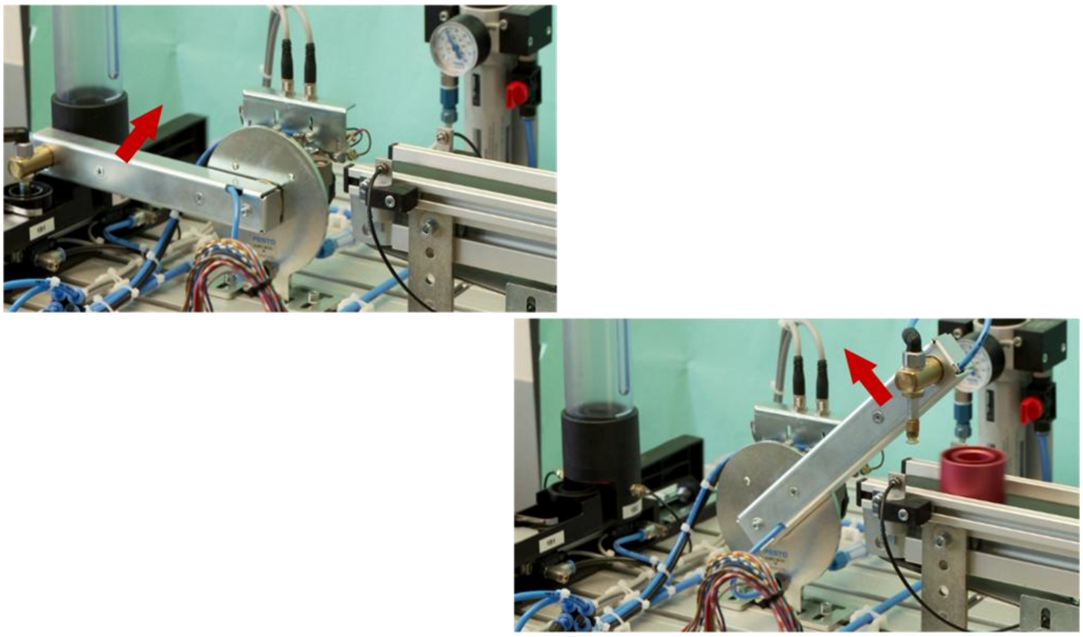


Figure 48: Transfer Arm

From basic trigonometry, 25mm of additional height requires approximately 10° of adjustment on both sides of the transfer arm i.e. from 175° to 165° to pick and from 60° to 70° to place the part as shown in Figure 48.

If the transfer arm is reconfigured in this manner, the workpiece can be lifted, transported and released to the conveyor belt as shown in Figure 49:

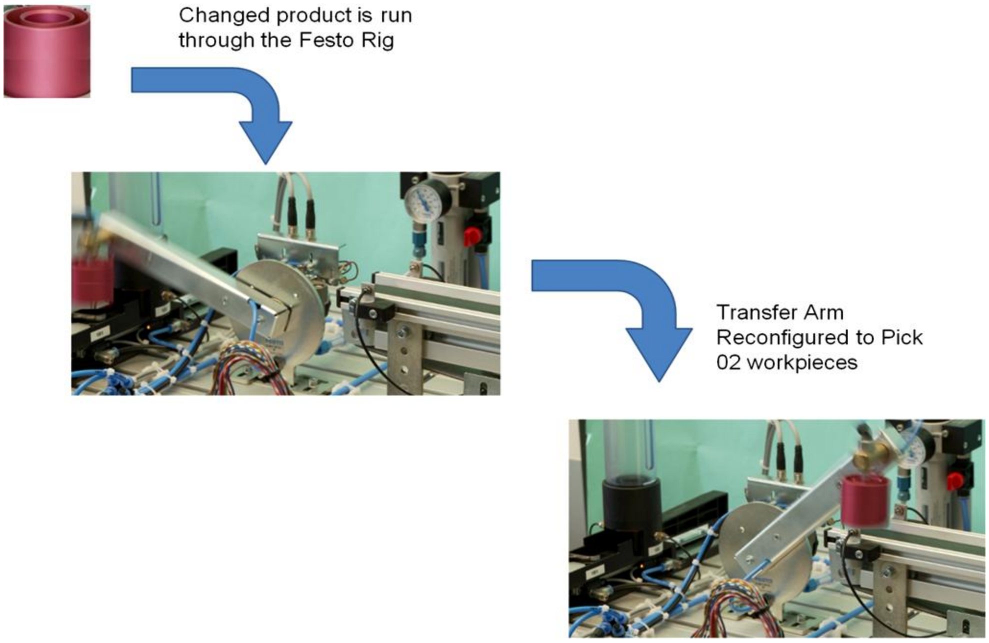

Changed product is run through the Festo Rig
Transfer Arm Reconfigured to Pick 02 workpieces

Figure 49: Transfer arm reconfiguration

The increased height (and weight) of the workpiece can be transported through the conveyor. The next station is the probing station. The changed workpiece can also be passed through probing station uninterrupted as shown in Figure 50.

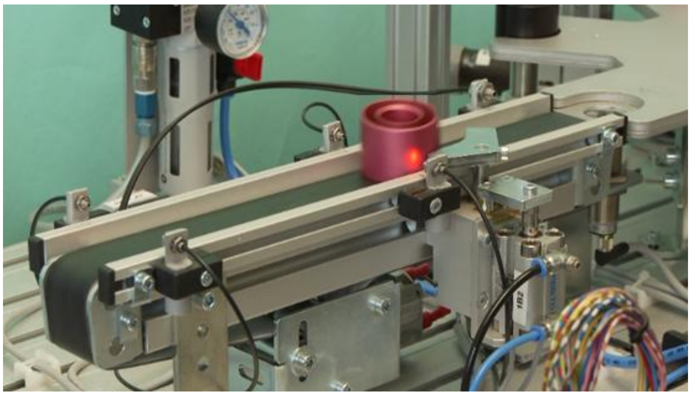


Figure 50: Punching unit reconfiguration correct here after

The next station is the punching station. The function of the punching station is to punch a hole in the workpiece. Here again the height handling capacity of the station is limited and the station needs reconfiguration. The workpiece would otherwise jam at this station due to the limited height of the station, as shown in the Figure 51. Again the rule of the height handling capacity of the punching station is added in the ontology.

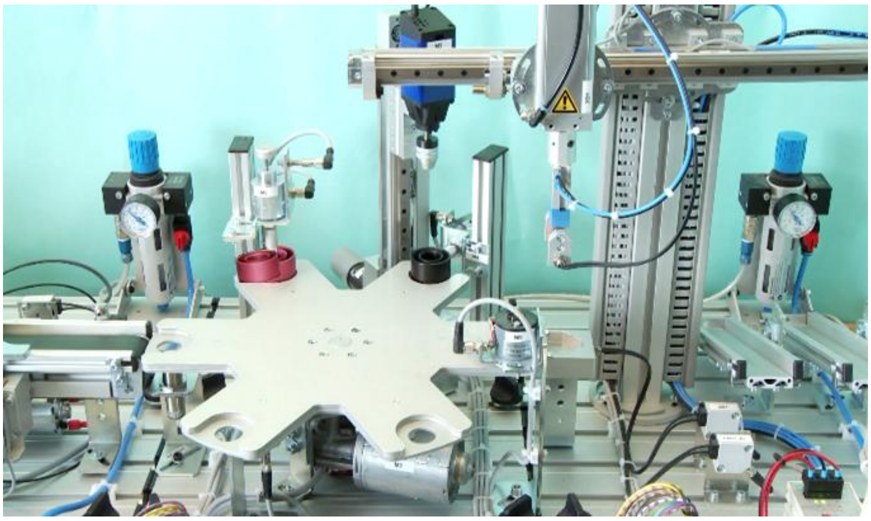


Figure 51: Punching unit reconfiguration

The next station is the drilling station, the drilling station has adequate height handling capacity and therefore, it only needs reprogramming as shown in Figure 52.

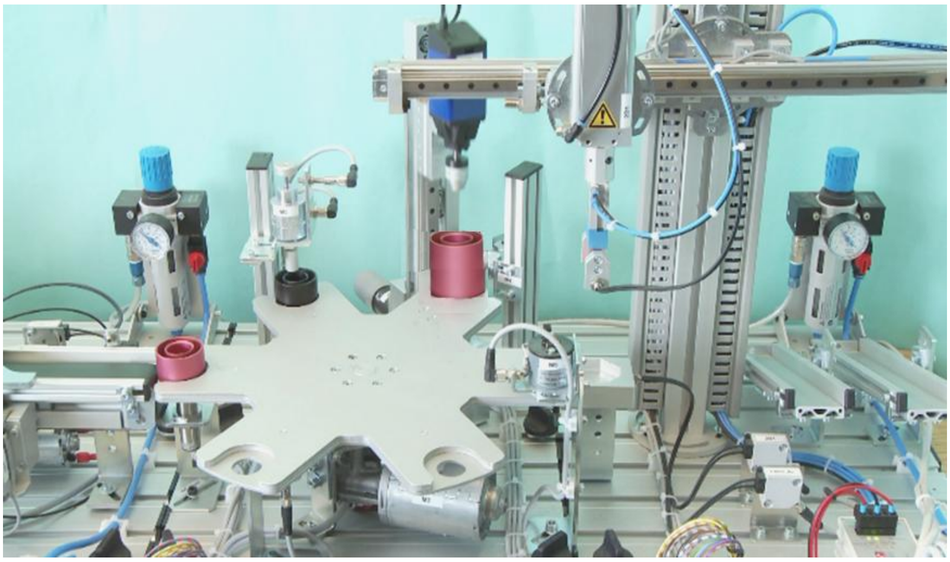


Figure 52: Drilling station reprogrammed

The next station is the handling arm, this also needs readjustment to cater for the increased height therefore reprogramming of the station is required as shown in Figure 53.

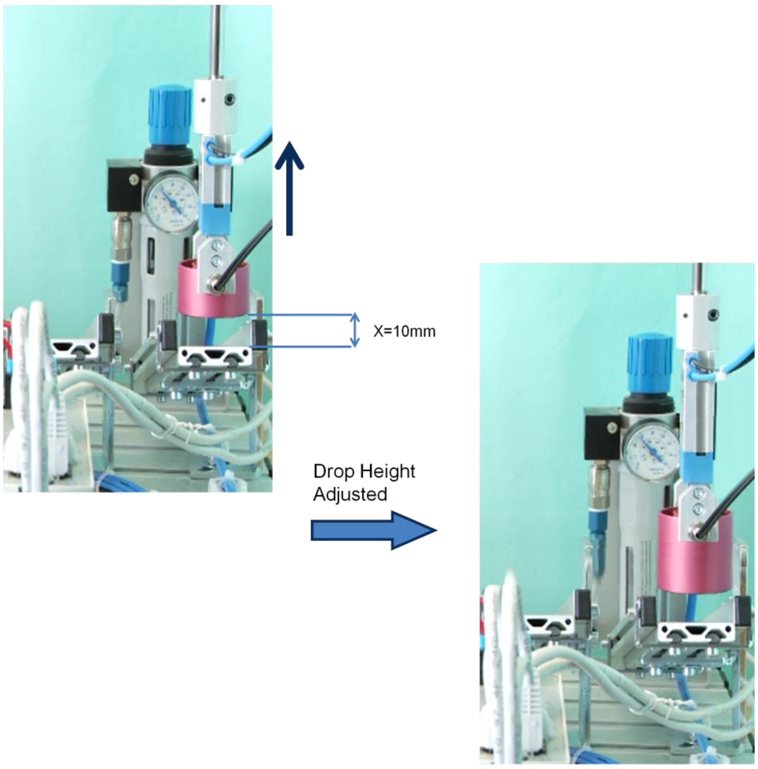


Figure 53: Handling arm height reprogrammed

The drop height (the position of the arm when it releases the workpiece) of the handling arm is 10mm approximately therefore it needs to be reprogrammed to cater for the increased height of the product as depicted in Figure 53. A summary of the rules and station retrofitting categories is provided in Table 14 below.

Workstation	Category of Change	Rules / Reason

Distribution Hopper	Redesign	*IF* workpiece height ≥ 28 *THEN* Redesign the station; *ELSE*
		IF {(workpiece 14<height < 28) *OR* (20<diameter < 40)} *THEN* Use-as-is the station
		*For the current test case, the workpiece was manually placed at the distribution hopper location

Transfer Arm	Reconfigure	Transfer Arm boundary positions need to be changed to accommodate increased height *IF* workpiece height ≠ 25 OR *IF* weight > 160gms *THEN* Reconfigure the station ELSE use-as-is If Diameter < 18 then Reconfigure

Conveyor	Use-as-is	*IF* workpiece weight > 160gm then Redesign *ELSE* use-as-is
		IF workpiece diameter > 45 then Redesign ELSE Use-as-is

Probing	Use-as-is	*IF* workpiece weight > 160gm then Redesign *ELSE* use-as-is (*IF* 45 ≤ product diameter ≤ 10 *THEN* Redesign station) ELSE Use-as-is
Rotary Table	Use-as-is	*IF* Diameter = 40 *THEN* Use-as-is *ELSE* (*IF* product diameter < 40 *THEN* Reconfigure station) AND (*IF* product diameter > 40 *THEN* Redesign)
Punching Unit	Reconfigure	*IF* product height ≠ 25 *THEN* Reconfigure station {Punching unit needs to accommodate greater height}
Drilling Unit	Reprogram	(*IF* product height ≠ 25 *AND* ≤ 60 *THEN* Reprogram height) AND (*IF* product height is > 60 *THEN* Reconfigure the station)
Handling Arm	Reprogram	(*IF* {product diameter ≠ 40 AND (55 ≤ diameter ≤ 20) *THEN* Reconfigure the station) (*IF* {product diameter ≠ 40 AND (20 < diameter < 55) *THEN* Reprogram the station) (*IF* product height ≠ 25 *THEN* Reprogram the station)

Table 14: Simple rules formulated on Festo rig

Table 14 summarises the results and it is observed that 37.5% of the components can be used without any change; while 25% require modifications in the control program only. Therefore

62.5% need no physical change in the components. Similarly 25% only require slight modifications i.e. the reconfiguration of the workstation or component of the workstation. Combining all the components with use-as-is, reprogram and slight modification requirements, it turns out to be 87.5% while only 12.5% need to be redesigned. Although this is only a very limited example, the results are consistent with the claims of Chung et al (2005) described in chapter 1, section 1.2.1, that typically 80% of the parts may be used without any change or slight modifications[76].

6.5 Festo Rig - Limitations & Lessons Learnt

Festo test rig provided the opportunity to analyse and evaluate some preliminary results. It showed that the concept is potentially viable and beneficial, but there were shortcomings with the ontology language used. In the OWL ontology, defining rules is not user friendly and the capacity of the OWL language to handle multiple rules is limited, as not more than seven rules can be handled in the Protégé OWL ontology.

From the product perspective, this limited scenario is also simplistic since, for example, on engine assembly machines, the product changes, as it is assembled, on almost every station. Another issue, likely to create semantical consensus and contextual heterogeneity issues later on, is the inconsistencies in terminology used in different applications. For example, workstations in the Festo rig training manuals are called ‘stations’ while the same items in the virtual models of the rig in CCE tool, are called ‘systems’. Similarly product and workpiece are defined seperately pointing to the same concept, this problem is widespread in the real world scenario. Therefore, it was realised to include data mediation capability in the developed system as well.

The development of the ontology for the Festo test rig helped to conclude that the concept of the research is applicable and useful, however, it proved necessary to switch to a more robust ontology langauge and investigate more valid real-world scenarios in engine assembly, where the product typically changes form at every workstation. Therefore, a new ontology language WSML (with WSMT editor) was tested and showed no problems with rule definition and number of rules. The WSML ontology language and its characteristics are explained in detail in chapter 2, section 2.6.1.2.

[76] Also corroborated by Ford Engineers at DTC, UK.

Rules vs. Fit Analysis

The rules formulated for the Festo rig proved useful when tested for changes in height and diameter of the workpiece. The iterative process of changing product dimensions and checking corresponding rules helped fine tune the rules. An iterative approach is recommended to be followed, where possible, so that a more refined version of rules may be obtained and confidence reached in the results. It is concluded that a rapid approach to product fit analysis can be realised through parameterised axiomatic process compared to a simulation-based virtual engineering tool. The repetitive analysis also provided an idea of how the change in dimensions of the product may affect the resource. Also the complexity of relational constraints becomes more evident, the product changes and retrofitting configuration is not a linear function thus broad, generic rules are hard to devise. Table 15 describes the example of the reiterative approach to analyse probable product changes and potential effects on the workstations.

Workstation	**Height Change**		**Diameter Change**		
	Standard case **2 x Height**	**½ Height**	**2 x Diameter**	**½ Diameter**	**¼ Diameter**
Distribution Hopper	Redesign	Use-as-is	Redesign	Use-as-is	*Reconfigure
Transfer Arm	Reconfigure	Reconfigure	Reconfigure	Use-as-is	Use-as-is
Conveyor	Use-as-is	Use-as-is	Redesign	Use-as-is	Use-as-is
Probing	Use-as-is	Use-as-is	Use-as-is	Reprogram	Redesign
Rotary Table	Use-as-is	Use-as-is	Redesign	Reconfigure	Reconfigure
Component Part Checker	Use-as-is	Use-as-is	Redesign	Redesign	Redesign
Punching Unit	Reconfigure	Reconfigure	Use-as-is	Use-as-is	Use-as-is
Drilling Unit	Reprogram	Reprogram	Use-as-is	Use-as-is	Use-as-is
Handling Arm	Reprogram	Reprogram	Redesign	Reprogram	Reconfigure
* The workpiece may wobble settling down in the distribution hopper with decreasing diameter					

Table 15: Rule refinement through iterative product change analysis

Plausible Product Profile Envelope

During the rule refinement process, it has also been observed that the change in product should be realistically planned in accordance to the practical scenario. For example, the change in diameter of the workpiece to quarter the original size apparently seems to have no drastic effect but as the Festo rig is not designed to handle very small (or conversely very large) workpieces, it might create unanticipated problems such as placing the product through distribution hopper can cause the workpiece to topple upside down or sideways. Thus the rules are only valid for a realistic product envelope.

6.6 Industrial Use Cases

Ford UK is involved in research, development, implementation and production of vehicle assembly operations, one of the most critical areas is powertrain (engine) assembly. Ford's Dagenham Engine Assembly Plant is the largest in the UK and also the largest producer of Ford diesel engines globally[77]. The plant is one of the most diversified and technically enhanced engine assembly plants with a capacity of 1.4 million engines/year, every 28 seconds a finished unit reaches the end of the line. It creates a wide range of mixed product diesel engines fitted to 28 different models of Ford, Jaguar, Land Rover, Volvos, Mazdas and Peugeot-Citroën that are destined for Europe, Japan, India and Brazil. The plant has the capacity to assemble different sizes and types of engines e.g. 1.4, 1.6, 1.8, 2.0, 2.2 and 2.4 litre alongside 2.7 and 3.0 litre V6 units and a 3.6-litre V8 engines, however, the majority of the current production consists of 1.4L and 1.6L diesel engines. The plant currently employs more than 4000 people. A snapshot of the engine assembly line at Dagenham plant is shown in Figure 54.

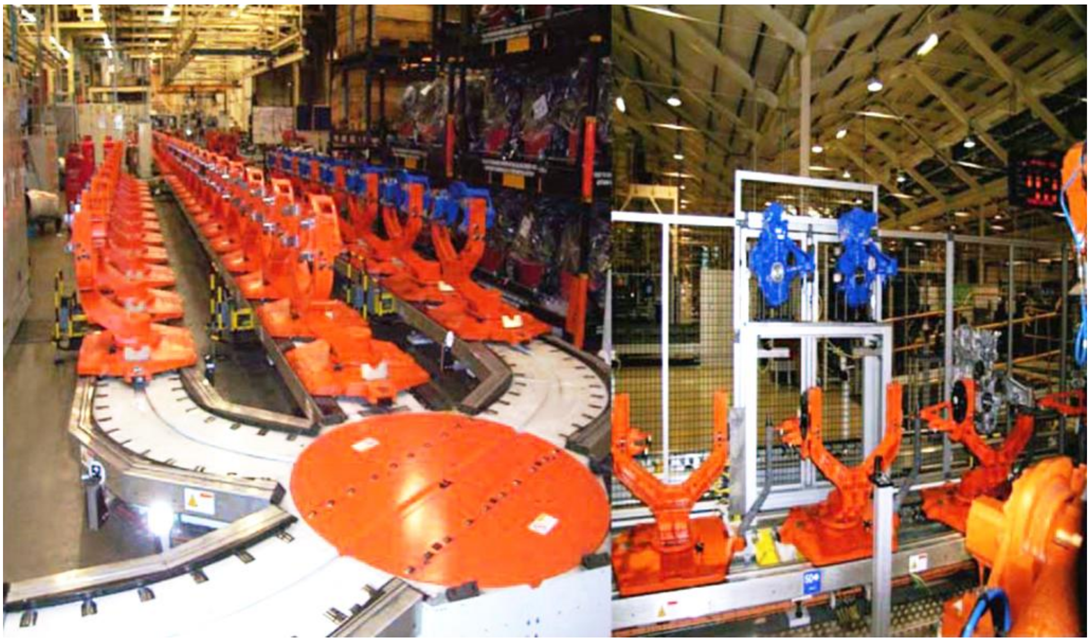


Figure 54: Snapshot of Tiger Assembly Plant at Dagenham, UK

A typical powertrain assembly line consists of approximately 150 workstations including fully automatic, semi-automatic and manual ones, each performing a specific assembly operation. The line is structured into smaller portions called zones, and the zones are composed of workstations.

[77] http://www.ford.com/technology/

6.7 Engine Fit Analysis

This section describes the purpose and procedure of the test cases inspired by the real industrial challenges in the automotive industry. The case study validates the conceptual framework and the prototype system providing solution to the stated problem. Engine assembly is a sequential process, several variants of an engine are assembled on a single assembly line. Each assembly stage of engine has key characteristics associated with the stations for that stage. Engine verification at each workstation will be carried out for a certain assembly stage of the product.

The aim of the case study is to partially automate the manual evaluation of PPR inter-dependency constraints in case of changes in the product, the summary is provided below:

6.7.1. Summary of the Case Study

Table 16 summarises the case study.

Engine Fit Analysis	
Industrial Scenario:	New engine or a variant of an existing engine is introduced onto the assembly line.
Requirement:	Rapid evaluation of effects of engine change on existing machines. There may be two situations i.e. if the changes required on the existing line are feasible then the changes will be incorporated. If there are changes on almost every station including major changes on most of the existing station then the new assembly line may be installed, either of the decision must be made quickly and confidently.
'As-Is' at Ford:	Manually appraise changes to each physical station
Why	There are approximately 150 stations on a powertrain assembly line and virtual verification through simulation / VBE / SBE is extremely time-consuming. The impact of product changes is not currently assessed by the end user (i.e. Ford) and machine design modification is controlled by the machine builder.
'To-Be' Approach:	Automatically appraise changes to each virtual station
How	Rapid constraint evaluation through rules and axioms. Critical decision variables are quantified in terms of KCs of products and the KCs are mapped directly to the machine capabilities. Thus effects of change of product features are linked to machines e.g. How does the depth of the oil pan sump affect the length of nut runners on OP1900?
Demonstration (Case Study 1) (Technique)	A real industrial case i.e. engine change from a three to four cylinder unit. Run the changed product through the entire assembly line to detect assembly hard points automatically through parameterised analysis.

Table 16: Summary of Case Study

A human expert will utilise all of the foregoing knowledge to come to a conclusion for engine fit analysis. For any change in part / parts of an engine, the first requirement is to retrieve all stations and components affected by the change. Every station performs one or many assembly process steps therefore the engine keep changing through addition/assembly of parts as it leaves every station. Therefore more than one part of an engine assembly is associated with a particular workstation and it is required to retrieve parts associated with other parts to make sub-assemblies (product set). Any change in one part of engine assembly does affect sub-assembly and may affect one or many stations and the first requirement is to retrieve all the associated engine parts as well as stations which are directly affected by the change. The next step is to check how these workstations are affected i.e. whether reprogram, reconfigure or redesign would be required on the affected stations. The next section describes a classic use case where an engine assembly changes from one design to another.

6.7.2. Business Requirement

Product change management necessitates holistic coordination of autonomously controlled entities. Automotive industry is struggling to acquire the ability to respond rapidly to product changes. Quicker and confident evaluation of effects of product change through rapid constraint evaluation is suggested to be the solution which would also help early verification of machines for precise investment costs.

A Real Industrial Case: To evaluate product change effects from 'Program Fox' (in-line three-Cylinder) to 'Program Sigma' (in-line four-Cylinder).

A typical use-case of converting an engine assembly line from handling a three-cylinder engine to also handling a four-cylinder engine with the existing approach and the suggested new approach is discussed. This is a real-world example which is considered a major change and this demands a complete run of the changed product to all the existing / generic workstations. In this scenario the KB system can be used to check engine fit possibility on each and every workstation by providing key characteristics of each product stage on a specific workstation.

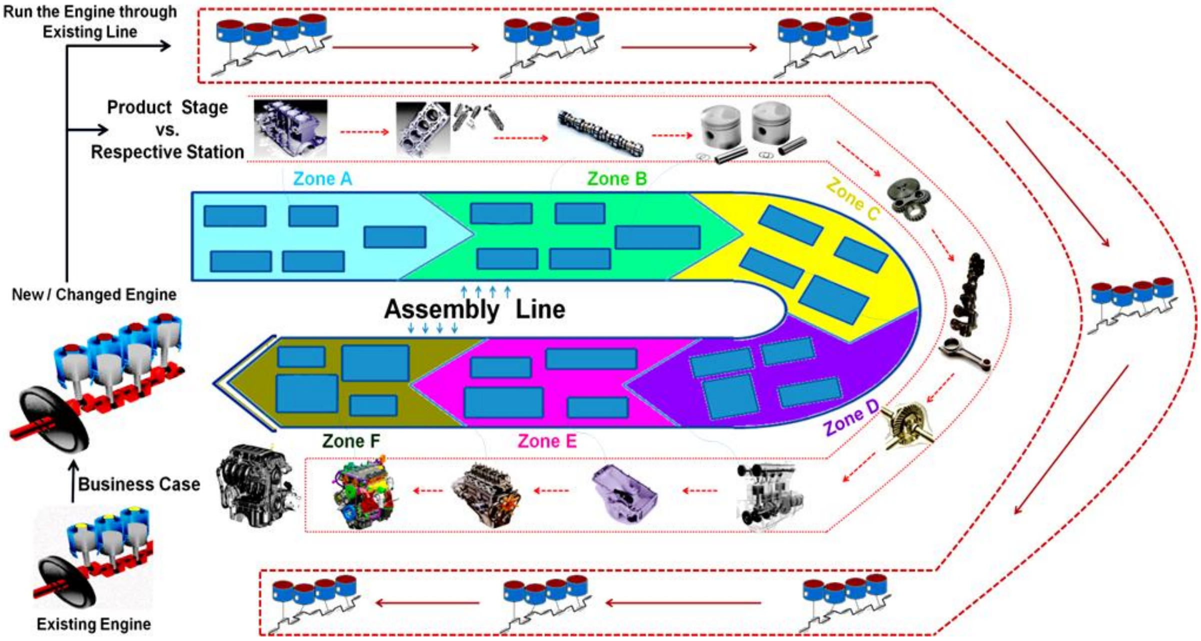


Figure 55: Engine change from 03-cylinder to 04-cylinder

The capability of the system thus needs to be extended to support an '*engine sub-assembly fit on respective station scenario*' as shown in Figure 55. As the engine keeps on changing on every station, so are the KCs of that assembly stage and the corresponding station capabilities. There are two fundamental requirements laid down under the stated use case (i) mapping among PPR and (ii) defining relational constraints. The ALDIMS tool can help evaluate effects of the changed engine on subsequent stations by making rules and axioms for assemblability. The rules and axioms are made up of KCs of product with respect to workstation and can provide accurate results if the respective KCs of a certain product stage are compared to the corresponding station on the assembly line.

6.7.3. Engine Fit Analysis Details

To emulate this industrial case, four stations on the Tiger assembly line are considered. (1) The first station, engine-mounting station i.e. OP60, (2) the RTV station, i.e. OP1860, (3) the next station in the sequence, the oil pan rundown station, i.e. OP1900 and (4) the very last station on the line, engine off-load station, i.e. OP2970, as depicted in Figure 56. These stations were selected after discussion with Ford engineers as these are most versatile stations containing complicated modules and carrying out sophisticated assembly operations.

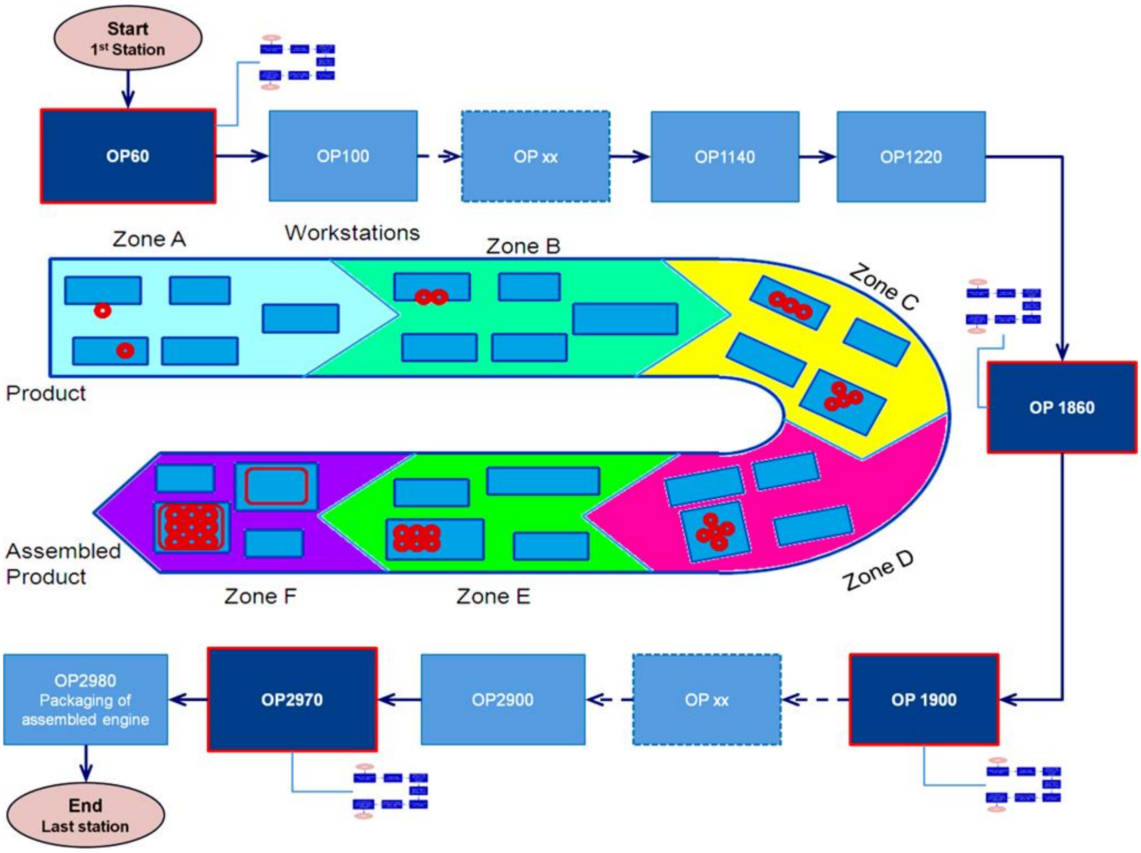


Figure 56: The selected four stations for case study from the DVM-4 powertrain assembly plant

The analysis of OP60 is described below. Stations OP1860, OP1900 and OP2970 are provided in Appendix 3 in order not to clutter the main text.

6.7.3.1 OP60 System – Engine Mounting Station

The block load station i.e. OP60 is the very first assembly process. The assembly process consists of mounting the engine block on the assembly line. This is a semi-automatic workstation. A snapshot of the OP60 station is shown in Figure 57.

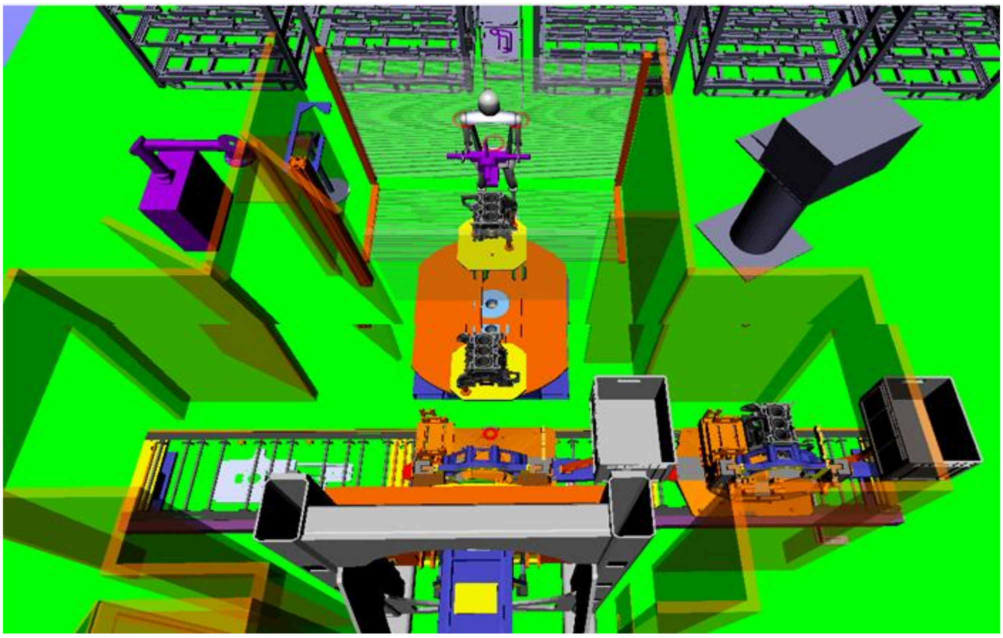


Figure 57: OP60 Station: Engine Mounting Station

The processes performed by OP60 are detailed in Figure 58. The cycle time of the station is 33.4 seconds.

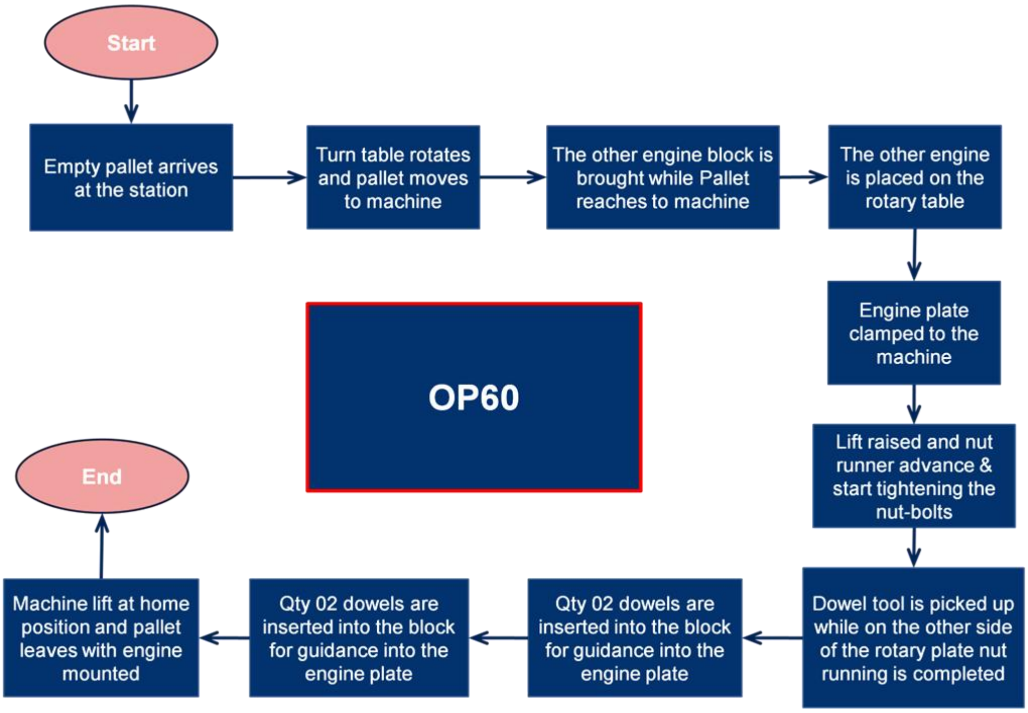

Start
Empty pallet arrives at the station
Turn table rotates and pallet moves to machine
The other engine block is brought while Pallet reaches to machine
The other engine is placed on the rotary table
Engine plate clamped to the machine
Lift raised and nut runner advance & start tightening the nut-bolts
Dowel tool is picked up while on the other side of the rotary plate nut running is completed
Qty 02 dowels are inserted into the block for guidance into the engine plate
Qty 02 dowels are inserted into the block for guidance into the engine plate
Machine lift at home position and pallet leaves with engine mounted
End
OP60

Figure 58: Assembly process sequence for OP60

The engine is mounted onto the engine plate which is already attached to the pallet at the specified position. The empty pallet arrives at the station, the operator will have already placed the engine block on the slide plate of the rotary table unit as shown in part 1 of the Figure 59. The rotary unit rotates clock wise by 180^o and the operator picks up another engine from the racks and brings it to the rotary table as shown in part 2 of the Figure 59. As the rotation is complete, the slide plate slides towards the platen at the specified position and operator places the engine on the rotary plate after rotation as shown in part 3 of the Figure 59. The engine block is lifted to the clamp height and nut runners start clamping the engine block onto the engine plate. Meanwhile the operator inserts dowel pins in the other engine as shown in part 4 and 5 of the Figure 59. The automatic operation is completed and the rotary unit is rotated to its original position, the pallet leaves the station and the next cycle is started as shown in part 6 of the Figure 59.

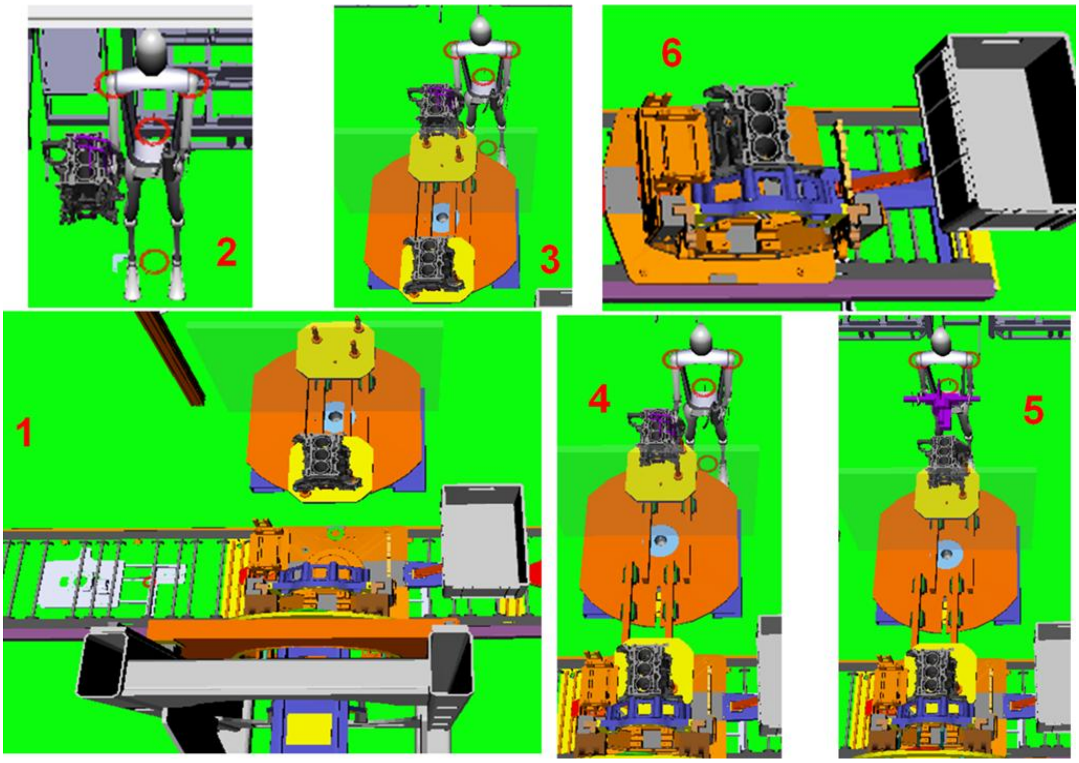


Figure 59: Simulation model of assembly station OP60 generated by CCE tool

The current assembly process consists of the engine block with three-cylinders. With the new business requirement, it is required to check the assemblability with four-cylinders i.e. a four-cylinder engine block.

The models of the three and four cylinder blocks on the OP60 station are shown below in Figure 60. The actual simulation process with the modeling tool with current infrastructure takes around

32 hours[78], therefore, it necessitates to get a quick evaluation of the possibility or otherwise of the stated business case without going into deeper details. This case is one of the recent new programs launched for the Ford vehicles where the engine changed from ‘three-cylinder Fox Program’ to ‘four-cylinder Sigma Program’. The diameter of the additional cylinder is exactly the same size as the existing ones, therefore, the major effect is on the width of the product i.e. X-axis of the machine, though other dimensions are also slightly varied. Therefore, the three-cylinder is replaced with four-cylinder block as shown in Figure 60.

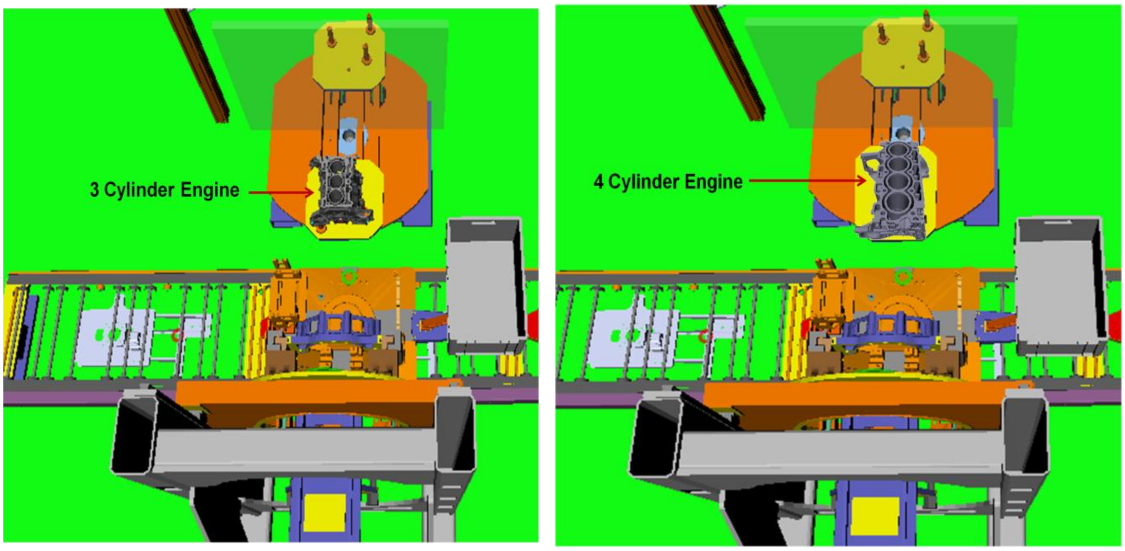


Figure 60: Simulation models 3-cylinder and 4-cylinder blocks in CCE tool

The first requirement is to confirm if the product has any potential hindrance along the three axes of the station. For this, the clearance space along X, Y and Z axes are defined in terms of product width, length and height. This allows possible variations allowed in the product along the axes of the machines without directly obstructing any module / component of the station.

To start investigating the engine change effects on the OP60 station, the first requirement is to align the four-cylinder engine with the three-cylinder engine. The Ford’s Product Engineers define an initial reference point of the engine block as (0, 0, 0), therefore, the reference points of the two blocks are matched and aligned and initially the two engines are overlapped to check if the position of the four-cylinder engine is aligned and matched with the exact positional reference of the three-cylinder block as shown in Figure 61. The changed four-cylinder engine block is

[78] Ford’s Fox powertrain assembly line virtual engineering time comparison exercise

coloured red to distinguish it from the existing three-cylinder grey coloured engine block. The reference point of the two engines is a green colour point as shown in Figure 61.

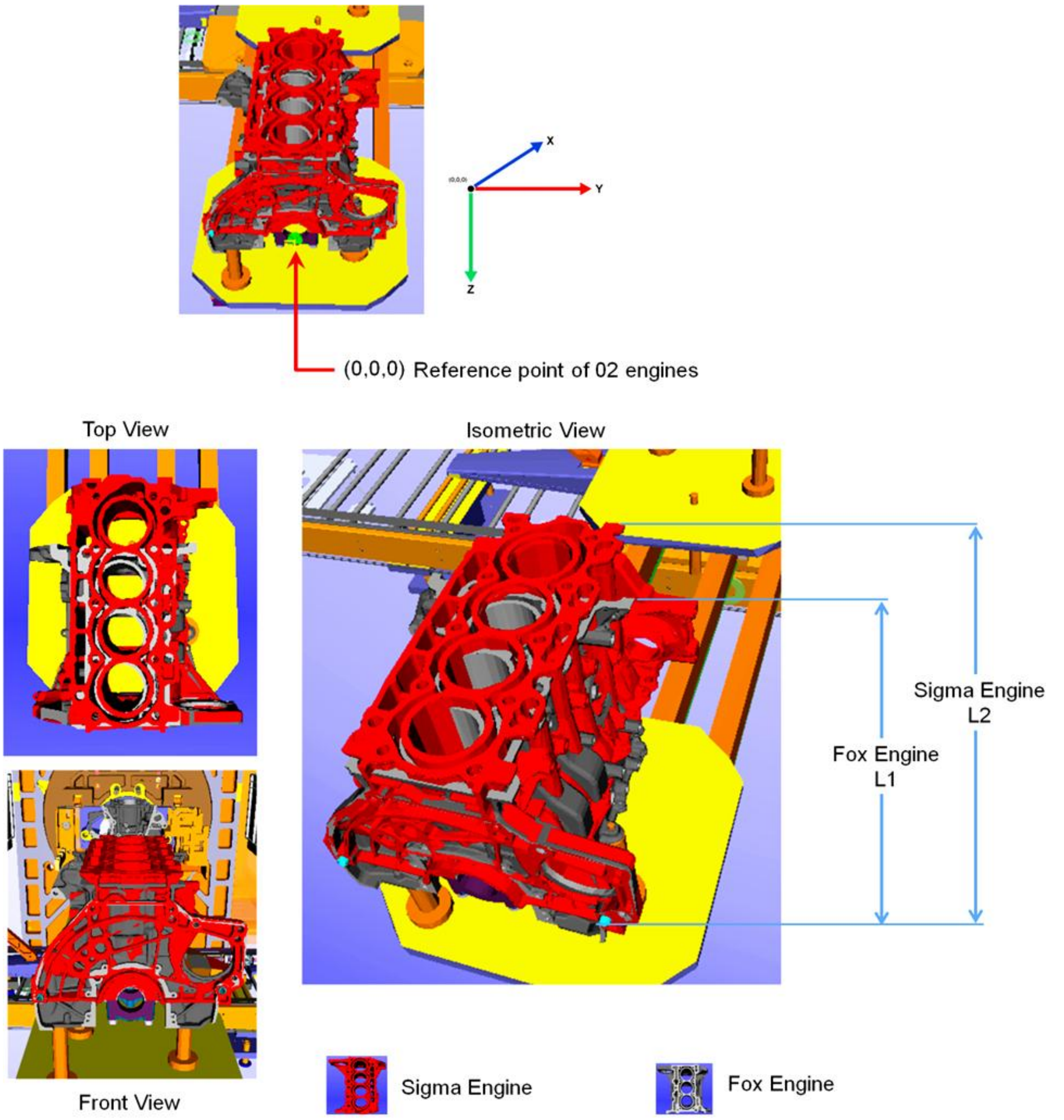


Figure 61: Overlapped engines of the two programs

The Figure 61 depicts slight variations in the two engine blocks, after matching and aligning reference points, other than the obvious variation in the width of the engine blocks (along the X-axis). The three-cylinder engine block is slightly larger than the four-cylinder engine block along Y and Z axes. The length and width are kept in consistent to the CCE convention.

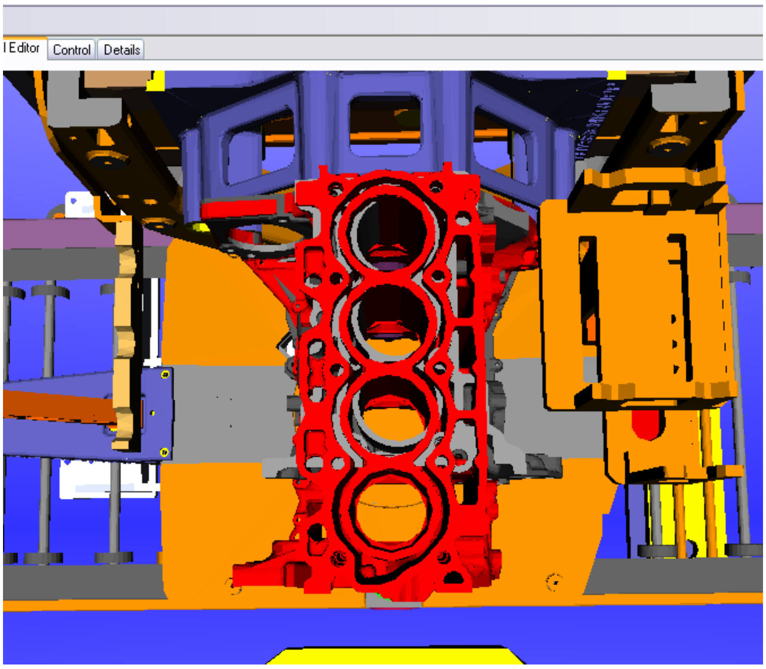


Figure 62: The two overlapped engines are run on the station

To evaluate the effects of engine change on the existing stations, the two overlapped engines are run through the station OP60 as shown in Figure 62. With the two overlapped engines on the reference point, the difference in dimensions can be easily measured. A few of the dimensions are stated below:

Three-cylinder engine block length = 280mm
Four-cylinder engine block length = 390mm
Three / Four cylinder diameter = 73mm (Bore = 63mm Wall Thickness = 4.5mm)
Three-cylinder engine block width = 215mm
Four-cylinder engine block width = 210mm
Three-cylinder engine block height = 260mm
Four-cylinder engine block height = 230mm

Once the two overlapped engines are run through the station for possible assembly hard points, the simulations are run independently for detailed analyses and defining simulation results in terms of specific axioms as shown below in the Figure 63.

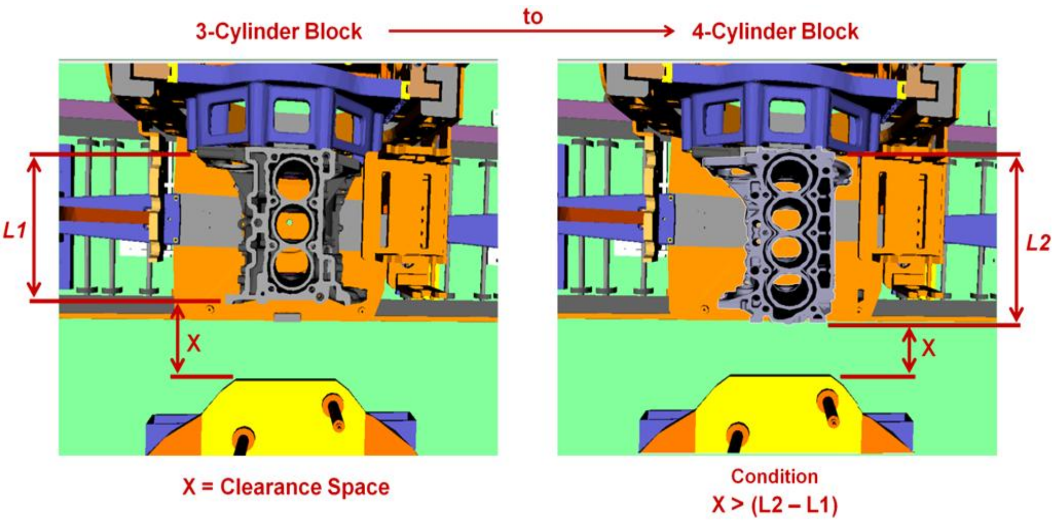


Figure 63: The clearance distance of the two blocks on OP60 station

The variable 'X' is the clearance space between the engine block and the slide plate on the rotary turn_table, while 'L1' is the length of the three-cylinder block and 'L2' is the length of the four-cylinder block as shown in the Figure 63. As the distance 'X' (X=150mm) in this case is greater than the difference in lengths of the cylinder blocks, therefore, the engine block can be placed on the platen without interference with the workstation.

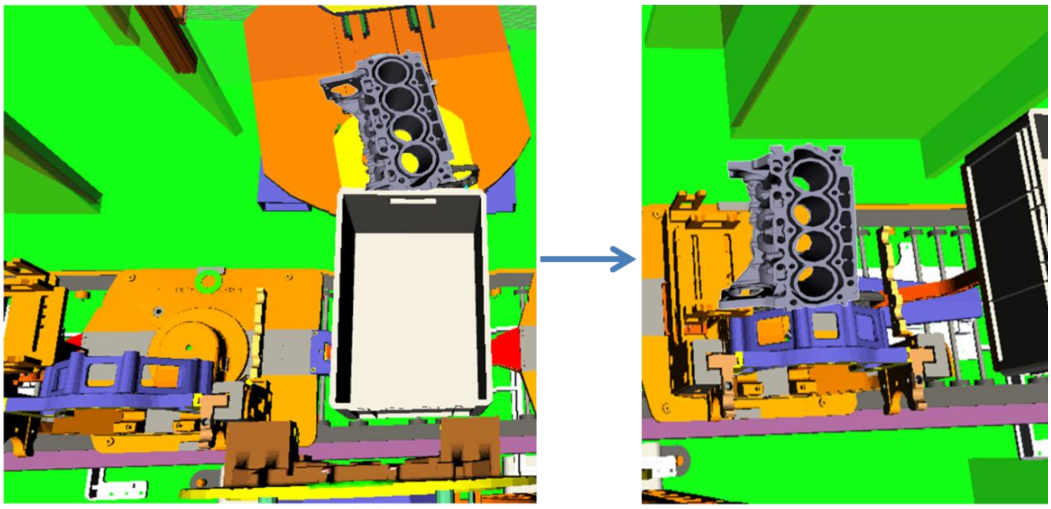


Figure 64: Simulation illusions from three to four cylinder engine change

The simulation shows a small gap between engine and one of the edges of the platen and replacing the three-cylinder engine block with the four-cylinder engine block shows an assembly hard point or a possible collision as shown in Figure 64. This is, however, not true as simulating the same process and viewing through a different angle (side view) reveals that the platen and rotating plate (and thus the engine block) are on different planes and there is no chance of collision. Apparently it is not readily deducible as it shows a crash between platen and the changed enlarged engine block. This requires extensive simulation from different perspective angles which transforms it into an extremely exhaustive and time consuming activity. However, the pinpoint problem is that after carrying out such a lengthy and meticulous activity, there is no infrastructure whereby the results / decisions may be documented. Therefore, this activity will have to be carried out again when there is any change in the engine or any part of the engine. The current research has addressed this issue and provided a basic infrastructure to record such decisions and observations in the form of rules and axioms.

Having confidently reached to the correct decision, the next step is to formalise the rest of the axioms. Cylinder block and cylinder head are made compatible so the requirement is to check clearance space and components functional capacities etc. The slight variations in four-cylinder engine block along Y and Z axes reveal that the existing station set-up can be used for the changed dimensions. However, the change in height needs to be taken into account. To keep the engine at the same height as a three-cylinder engine block there are two possible options i.e. (i) either increase the length of the resting pads on the engine slide plate or reduce the length of the platen. Platen is a common component for all stations on the line, therefore, every effort is made to keep the common components unchanged hence the best option is to increase the height of the resting pads on the slide plate. Similarly the location of the resting pads also needs to be changed according to the four-cylinder engine block design. The height and location of the resting pads for the three-cylinder engine block is shown in Figure 65.

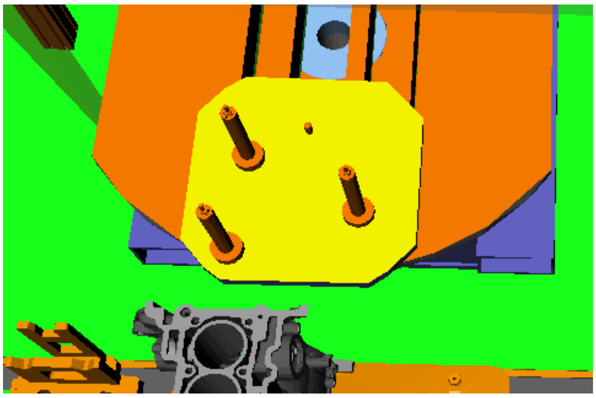


Figure 65: The engine resting pins on sliding plate for three-cylinder engine block

Engine resting points for the four-cylinder engine are different and the height is also different, therefore, the resting pads need to be repositioned and reconfigured so that the same engine plate and the same pallet can be used.

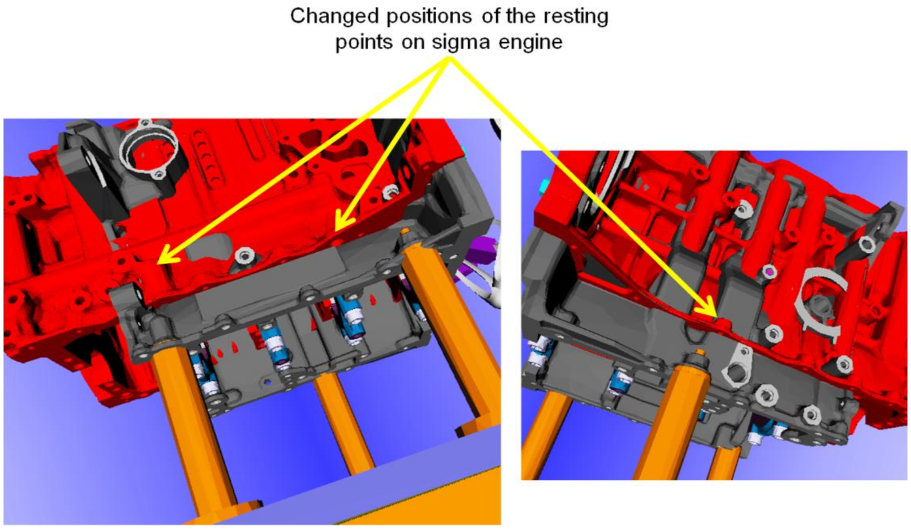


Figure 66: The resting pads on the four-cylinder engine are different to the existing positions

The Figure 66 shows the difference in height and location of the resting pads as are required for the four-cylinder engine. The Table 17 summarises the KCs defined for the OP60 station.

KC	ofType	Property	Existing value	Changed value
OP60_Nut_Runner_Bolts_Qty	Integer	hasValue	4	4
OP60_Nut_Runner_Diameter	Integer	hasValue	6mm	6mm
OP60_Nut_Runner_Length	Integer	hasValue	20mm	20mm
OP60_Nut_Runner_Torque	Integer	hasValue	10Nm	10Nm
OP60_Dowel_Pin_Diameter	Integer	hasValue	20mm	20mm
OP60_Dowel_Pin_Distance	Integer	hasValue	150mm	150mm
OP60_Engine_Slide_Plate	Integer	hasValue	Engine LxWxH	Changed Engine LxWxH
OP60_Product_Length	Integer	hasValue	280mm	390mm
OP60_Product_Width	Integer	hasValue	215mm	210mm
OP60_Product_Height	Integer	hasValue	260mm	230mm
Engine_Version	String	hasValue	“1.4L”	“1.6L”

Table 17: The KCs of the product on OP60

The Table 18 summarises the KCs, existing and changed attributes for the KCs as well as rules and axioms defined on the KCs for the OP60 resource.

KCs	Rules / Constraints	Existing attribute	New / Changed attribute
	Rules		
OP60_Cylinder_Block Major change	IF $X >$ (L2-L1) THEN check for product-machine clearance and tooling capabilities If corresponding changes in cylinder head have been incorporated Then check for clearance spaces and tooling capabilities, the system will check for the product-resource constraints	3-Cylinder	4-Cylinder

OP60_Engine_Mounting_Plate Dowel pins diameter Dowel pins distance	IF ?dia!=20 THEN Redesign Dowel_Pin ELSE IF {?dia !=20 and (?dia≥18 and ?dia ≤ 22) and ?distance=150} THEN Reconfigure Engine_Mounting_Plate ELSE IF (?distance!=150) THEN Redesign Engine_Mounting_Plate	20 150	20 150
OP60_Nut_Runner Qty Diameter Length Torque	If ?qty = 4 AND {(?dia ≥ 6 and ?dia ≤ 12.5) or (?length !=20 and ?length ≤ 38) AND (8≤?torque ≤ 11)} THEN Reprogram NutRunner ELSE IF ?qty = 4 AND {(25.4≥?dia > 12.5) or (50.4≥?length >38) or (14≥?torque >11)} THEN Reconfigure NutRunner ELSE IF [?qty = 4 AND {(?dia > 25.4) or ?length >50.4 or ?torque >14} OR ?qty!=4] THEN Redesign NutRunner	4 6 20 10	4 6 20 10

OP60_Engine_Slide_Plate
Mounting pads
	If length of the engine block changes then readjustment is required in engine slide plate.
	IF change in length < 150 i.e. engine block length L < 430 then reprogram slide distance
	ELSE IF 430<L<560 THEN Reconfigure
	ELSE IF L > 560 THEN Redesign the slide plate
	AND
	IF width/length/height of the block changes then readjustment of mounting pads / resting pins may be required in terms of Height / Location
	IF position of pads/ pins ≠ existing value THEN Reconfigure / Redesign the pads / pins
			Length=390
			Width=210
			Height=230

Product length x width x height
(clearance in x, y, z)
Product Height - Z-axis
Length - X-axis - man & m/c distance
Width - Y-axis - Along Rail Travel
	IF (?length > 280 and ?length < 450) AND ?width ≤ 250 AND ?height<320 THEN Use-as-is System
	ELSE IF (?length ≥ 450 and ?length ≤ 500) AND (?width > 250 and ?width < 300) AND (?height>320 and ?height< 450) THEN ReProgram System
	ELSE IF (?length > 500 and ?length < 630) and (?width ≥ 300 and ?width ≤ 450) and (height>450 and ?height ≤ 650) THEN Reconfigure System
	ELSE IF (?length > 630) or (?width >450) or (?height>650) THEN Redesign System
		Length=280
		Width=215
		Height=260
			Length=390
			Width=210
			Height=230

OP60_Pallet (Common component)	In the current engine change: Use-as-is {The pallet may need to be changed for the sigma engine if the resting pins are not changed to adjust the height of the new engine, however, as redesigning a pallet is economically not viable, therefore, it is suggested to change the resting pads}	The pallet changes depend upon engine mounting plate, it is tried to change mounting plate and keep the pallet unchanged	
OP60_Rail (Conveyor) (Common component)	In the current engine changed: Use-as-is {Conveyor reconfiguration / redesigning is most expensive, therefore, every effort is made to keep the conveyor unchanged}	Rail carries the platen therefore if platen is unchanged then the rail is unchanged	
Engine Mounting Plate (Common component)	In the current engine changed: Use-as-is {Again, it is desired to keep the engine mounting plate unchanged unless absolutely necessary}	If the platen is unchanged, engine mounting plate will also be unchanged provided nut runners & dowel pins are same	

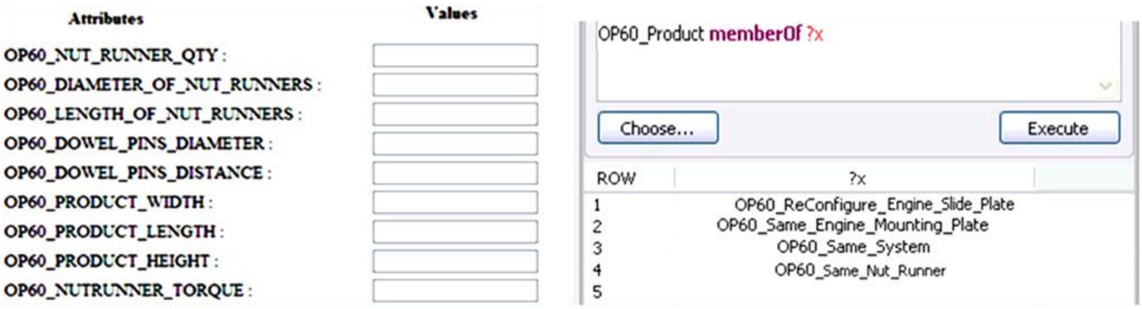
Table 18: Rules and axioms formulated on the KCs of OP60 System

As the attributes are supplied to the system, the output is provided with recommendations of retrofitting categories for the OP60 station as shown in Table 19.

OP60 (Engine Mounting Station)			
Input (attributes and values)		**Output**	
		Machine / Component	**Required Action**
OP60_Nut_Runner_Bolts_Qty	4	Nut Runner	Same
OP60_Nut_Runner_Diameter	6		
OP60_Nut_Runner_Length	20		
OP60_Nut_Runner_Torque	10	Engine Mounting Plate	Same
OP60_Dowel_Pin_Diameter	20		
OP60_Dowel_Pin_Distance	150	Engine Slide Plate	Reconfigure
OP60_Product_Length	390		
OP60_Product_Width	210	System	Same
OP60_Product_Height	230		

Table 19: Summary of Inputs and outputs for OP60 in ALDIMS

The user can access the user form (JSP web-interface form) through the website and can provide the input values, a snapshot of input parameters and output from the ALDIMS is shown below in Figure 67.



Attributes	Values
OP60_NUT_RUNNER_QTY :	
OP60_DIAMETER_OF_NUT_RUNNERS :	
OP60_LENGTH_OF_NUT_RUNNERS :	
OP60_DOWEL_PINS_DIAMETER :	
OP60_DOWEL_PINS_DISTANCE :	
OP60_PRODUCT_WIDTH :	
OP60_PRODUCT_LENGTH :	
OP60_PRODUCT_HEIGHT :	
OP60_NUTRUNNER_TORQUE :	

```
OP60_Product memberOf ?x
```

ROW	?x
1	OP60_ReConfigure_Engine_Slide_Plate
2	OP60_Same_Engine_Mounting_Plate
3	OP60_Same_System
4	OP60_Same_Nut_Runner
5	

Figure 67: Input KCs and the query output result by the KB system

The system has provided a means to record valuable information deduced from the extensive simulation activity in the form of rules and axioms. In this way the knowledge of the existing program is used to make quick decisions for the new program. Similarly, these decisions are recorded which again may be used for any change in the existing engine or a new engine program.

6.7.3.2 OP1860 System – RTV station

6.7.3.3 OP1900 System – Oil Pan Run Down station

6.7.3.4 OP2970 System – Engine offload station

Stations ‘OP1860’, ‘OP1900’ and ‘OP2970’ are also tested in the same manner as ‘OP60’ and the detailed results are provided in Appendix 3.

6.7.4. Analysis of the Case Study

The industrial case study has provided an opportunity to critically analyse the ALDIMS application, a detailed analysis and evaluation is presented in chapter 7, however, an appraisal of the system in terms of the engine fit analysis is described here. The current system has provided an infrastructure to capture, document, utilise and apply knowledge. The implicit knowledge is transformed into rules and axioms, a total of approximately 125 rules have been formulated on the four most diversified and complex stations. The workstation retrofitting is categorised into four groups, even though the engine change from three-cylinder to four-cylinder is considered a major change, yet many of the existing resources may be utilised and approximately only 20% need to be redesigned. The system provides a readily available solution to the process engineers as to what needs to be simulated and virtually verified. The system and its modules / components which are declared under the category ‘*use-as-is*’ and ‘*reprogram*’ need not to be virtually verified and can be passed straight on to the machine tool builders. On the other hand, the stations and its smaller units which fall under the category ‘*redesign*’ and ‘*reconfigure*’ are required to be modelled and simulated as a priority task.

6.7.4.1 Knowledge-based Line Design

The knowledge is captured and transformed into decision making platform in the shape of a knowledge base system thus, the current system has provided a knowledge based support in line designing / reconfiguring process e.g. the oil pan run down operation, OP1900, the detail is provided in Appendix 3, after the application of the ALDIMS tool is shown below in Figure 68.

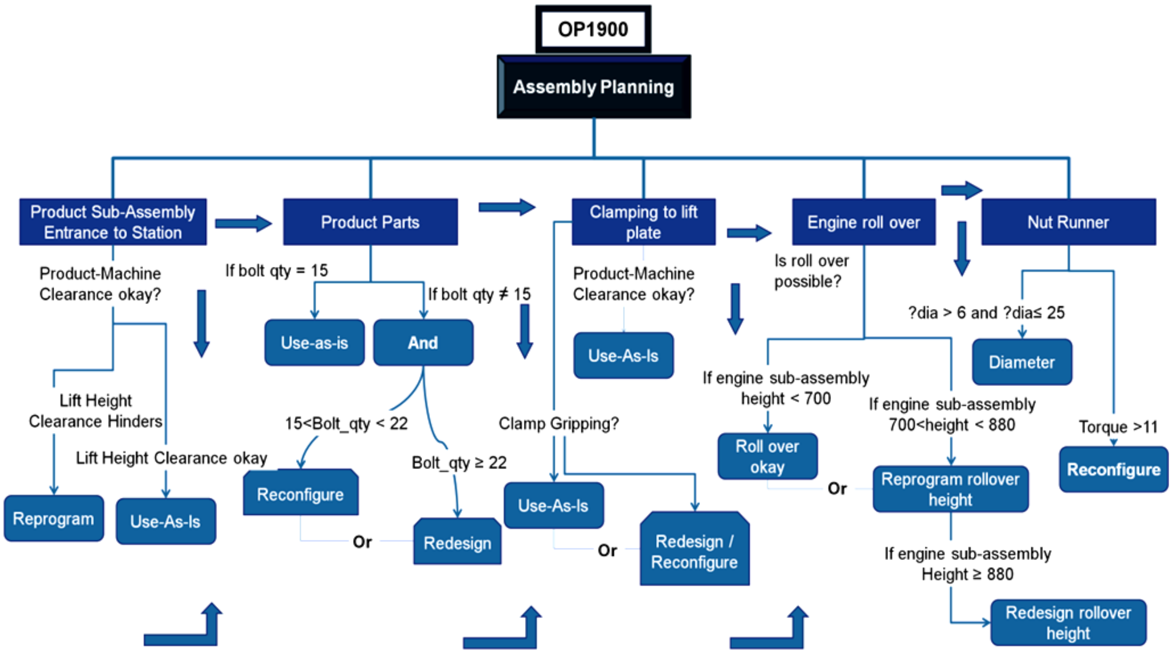


Figure 68: Knowledge based assembly line design

The Figure 68 represents part of the OP1900 system design activity i.e. taking into account only PPR entities. The design / reconfiguration activity consists of complex decision making at each and every step along the system design. The framework emulates the human thought process and implicitly documents each decision making activity. These steps provide the decision support for each distinctive design task from entry of the product sub-assembly into the station's proximity till the exit out of the station. The knowledge based system provides a relational constraints scenario to make decisions along the design of workstations. It was also revealed that not only products and machines are directly related to each other but one product part may affect another part / sub-assembly as well.

For example, change in volumetric capacity of oil pan sump is directly affecting the robots, clearance spaces, etc. This change also affects the length of the bolts to be used for tightening the nuts which is a product part. Similarly, the depth of oil pan sump does not directly affect the RTV station instead reprogramming the height of robot / adjusting the fixture can solve the change effect, however, the depth attribute has a much deeper impact on the next i.e. OP1900 station which is difficult to anticipate without a knowledge based support. Therefore, the system has transformed implicit relations and rules into explicit constraints. This has either eliminated or reduced the simulation efforts.

6.7.4.2 Knowledge Reuse Infrastructure

Presently these decisions are being made on the expert judgement of automotive domain engineers, tiresome virtual / slow build events or through tedious and time consuming simulation of 3D models of the product. Due to lack of any infrastructure for recording the decisions in a reusable form, these time consuming activities cannot be translated into knowledge and the outcome, though documented, cannot be reused, thus it is lost. The current KB system has provided an infrastructure to transform the simulation / slow or virtual build events into axiomatic arguments and records the outcome and decisions in a reusable form thus avoids such repeat cycles for the next change. The decision variables of KB input activities are quantified in terms of KCs of products/tools and these KCs are mapped directly to the machine capabilities. Therefore, results of one simulation activity or slow/virtual build event can be used for next engine programs, thus, avoiding repetition of the same activities and duplication of efforts in case of frequent changes in powertrain systems.

6.7.4.3 Line Design Control and Cost Analysis

The automotive OEMs currently bear the cost of the complete build of the assembly line. As stated before, any new assembly line consists of approximately 80% of the existing design / slight modifications. As the line design is not in the control of automotive OEMs, therefore, the cost of the complete assembly line is borne by the OEMs. The BDA project and the current research aim to change this practice. The automotive OEMs need to bear the cost of changes in the existing assembly line. The current research would help rapid evaluation of the total changes required i.e. 'Δ', in the cloned systems and in complete assembly line.

New Assembly Line = Existing Assembly Line Cloned $+ \Delta$

Δ = Changes / Amendments Required = New requirements – Existing solutions

The developed system can quickly evaluate the changes required in the line, therefore, the order will be placed only for the changed resources / components of the resource rather than the complete line order. In this way the system will help the automotive OEMs to take control of the machine design and cost analysis activity.

6.7.4.4 Limitations

The ALDIMS application provides decision support, constraint analysis and decisions management in the automation resources' virtual verification process only. Assembly line design involves several closely related activities e.g. quality control, workstation layout, productivity analysis, ergonomics study which are currently outside the scope of the system. Similarly in the constraint analysis, some features of product / resource cannot be easily converted into numerical values. For example, the profile of the oil pan sump on the OP1860 workstation (detail of OP1860 workstation is provided in Appendix 3) has no specific value, therefore, it cannot be transformed in the form of rule / axiom. This is one of the limitations of the current system. The profile of the oil pan sump and the matching product sub-assembly is shown in Figure 69.

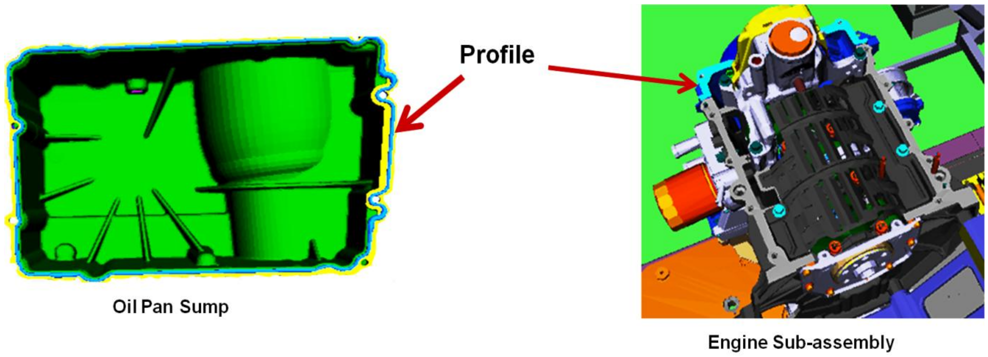
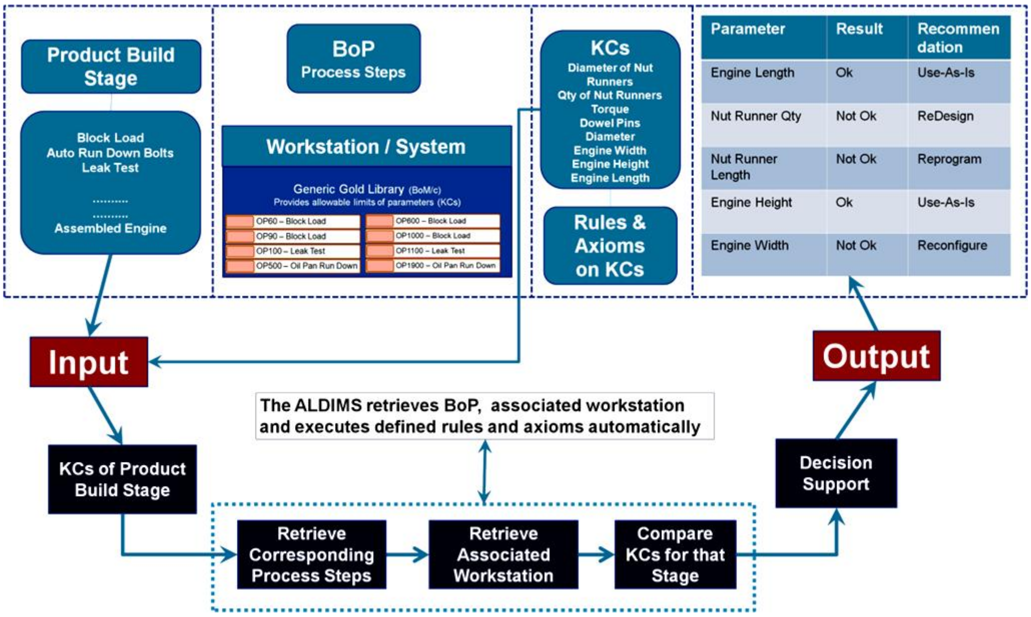


Figure 69: Typical profile on the oil pan sump and engine sub-assembly

The maximum and minimum points of the profile in X and Y axes, along with several intermediate consecutive points, can be compared with the corresponding values of the changed product profile to see the difference between the two, currently this is left for the future enhancements.

6.7.5. *Conclusion of Case Study*

The case study has shown that the knowledge based support has the potential to provide a rapid evaluation of the product-resource constraints (through process steps). The knowledge based rules are formed by actual simulation by an expert engineer. The current system has provided a basic infrastructure to record the decisions for later programs. The results of the KB system are reliable enough to be used for intrigue decision making, however, potentially problematic stations may still required to be further analysed e.g. detailed simulation analysis. For every change in the existing engine, it is not possible to carry out detailed simulation analyses on all the stations of the assembly line, therefore, it is suggested to use relational constraints evaluated from previous simulation analyses, thus, a tool for knowledge reuse and decisions management. Therefore, a platform is provided to conserving and documenting decisions, consequently, the assembly planning transforms from resource-based view to knowledge-based view by recording decision making conditions and translating the results of simulation analyses to imperishable knowledge. The engine fit analysis is parameterised, simulation and virtual verification is reduced and decision support provided thus the system support speeds up the overall line design / retrofit activity. The input, output and sequence of steps are summarised in Figure 70. The user will provide KCs of the changed product for a certain build stage as input and the KB system will provide decision support in terms of changes required on the existing resource i.e. the effect of the new parameters of the product on the existing workstation.



Parameter	Result	Recommendation
Engine Length	Ok	Use-As-Is
Nut Runner Qty	Not Ok	ReDesign
Nut Runner Length	Not Ok	Reprogram
Engine Height	Ok	Use-As-Is
Engine Width	Not Ok	Reconfigure

Figure 70: Engine fit analysis – flow diagram

The KB system will retrieve process steps, associated resource and compare new KCs against existing KCs, these three steps shown inside dotted rectangle, will be completed automatically.

6.8 Support Activities Introduced

In assembly line design / reconfiguration scenario, a combination of the functionalities are required to help realise engine fit analysis, these include, data mediation, efficient selection of resources and knowledge visualisation through hierarchical classification techniques. The next sections describe these support activities introduced in the developed system.

6.8.1. Support Activity 1: Data Mediation

The input and output of the ALDIMS tool may come from and go to autonomous, heterogeneous and distributed applications with syntactical and semantical dissimilarities. This issue is addressed in the ALDIMS tool to carry out engine fit analysis smoothly regardless of the structural / conceptual differences in software applications, changes in supply chain partners and even synchronisation of the KB system for changed automotive OEMs.

6.8.1.1 Summary of the Support Activity 1

Table 20 summarises the support activity 1 introduced in the ALDIMS.

Support Activity 1: Data Mediation	
Industrial Scenario	Different supply chain vendors with terminology mismatches Different applications with localised terminology
Why	Disparate application and distributed teams often result in conflicts in the structure and semantics of the concepts/data.
Requirement	Smooth flow of information across applications with semantic harmony
'As-Is' at Ford	Semantic / Syntactic heterogeneity in applications No data mediation facility, semantic mismatches are dealt manually
'To-Be' Approach	Unambiguous communication through automatic mediation among applications and supply vendors
How	To facilitate interoperability among semantically heterogeneous concepts of localised applications within Ford and among supply chain vendors through data mediation. To harmonise the data so that all the parties understand same semantics of the localised terminologies without any changes to the conventional vocabulary and concepts / language translation barriers. With the help of data mediator layer, same concepts defined differently can be retrieved while different concepts defined similarly can also be distinguished with a single query.
Demonstration	Mapping of ontology files Mediated query

Table 20: Summary of support activity 1

Ford Motor Company being the global company is spread all over the world. It operates vehicle assembly plants all over the world, including locations in USA, UK, Germany, Spain, Belgium, Romani, Mexico, Brazil, Venezuela, India and others. It has product design teams, process planning / optimisation teams and machine design and feasibility study teams over different corners of the world with different languages, representations and terms for same concept. Similarly, Ford's supply chain vendors e.g. machine and control vendors are spread all across the globe exacerbating the stated issue. For one program, the machine builder may be a German OEM, another with Italian and a further program may have Japanese suppliers.

The aim to create an integrative environment for PPR domains cannot be achieved without data mediation among software applications as well as among distributed supply vendors. The outcome of globalisation, on one hand, and OEM specific localised software applications, on the other hand, demand data mediation techniques. In the current case study, the Ford's supply chain vendors e.g. machine builders and control vendors are spread all across the globe. In the case study of two different programs i.e. from three-cylinder to four-cylinder engine, if it is decided to build a new assembly line due to large number of reconfiguration / redesign effort, and the new line is vendored out to a new machine builder, a major hurdle faced by engineers is mismatches in the prevalent vocabulary / terms. Two different engine programs may be designed by Ford USA, contains machines from different suppliers and installed at different locations. For example the Fox engine is designed by Ford USA, contains machines from Camau SpA Italy and installed in Romania while the Sigma engine is designed in USA, machine suppliers from Krause GmbH Germany and installed in Brazil and India. A data mediator addresses terminological mismatches that may occur among concepts, properties or services due to continuing variation along applications, teams and languages. Ontology mediation model applied in the current scenario is shown in Figure 71.

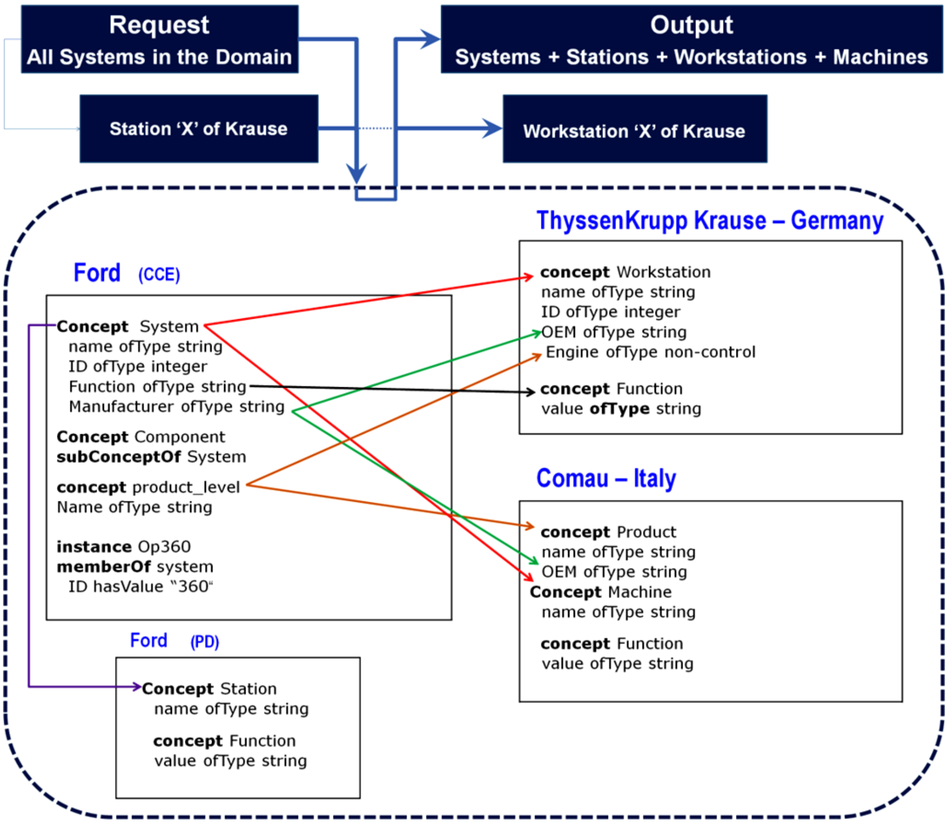


Figure 71: Mediation requirements and conceptual mappings in ontologies

Data mediators help instance transformation by translating from one concept (source) to the other one (target). Therefore system station and machine which are used inter-changeably at different Ford sites are mapped and mediated through ontology. During the course of the current research study, the author has had the first hand experience and associated problems for two different engine programs having separate machine builders with different production sites where the need for data mediation becomes ever more significant. This necessitates the requirement to harmonise the data so that all the parties as well as machines understand the same semantics of the localised terminologies without any changes to the conventional vocabulary and concepts. Therefore all the stakeholders have the need for an integrated, interoperable yet autonomous working environment as depicted in Figure 72.

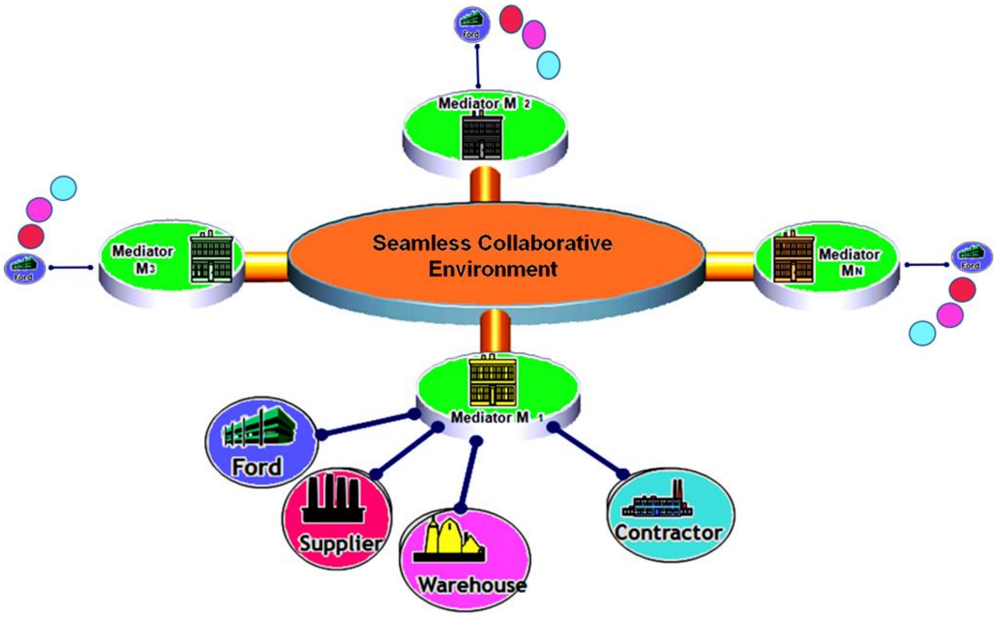


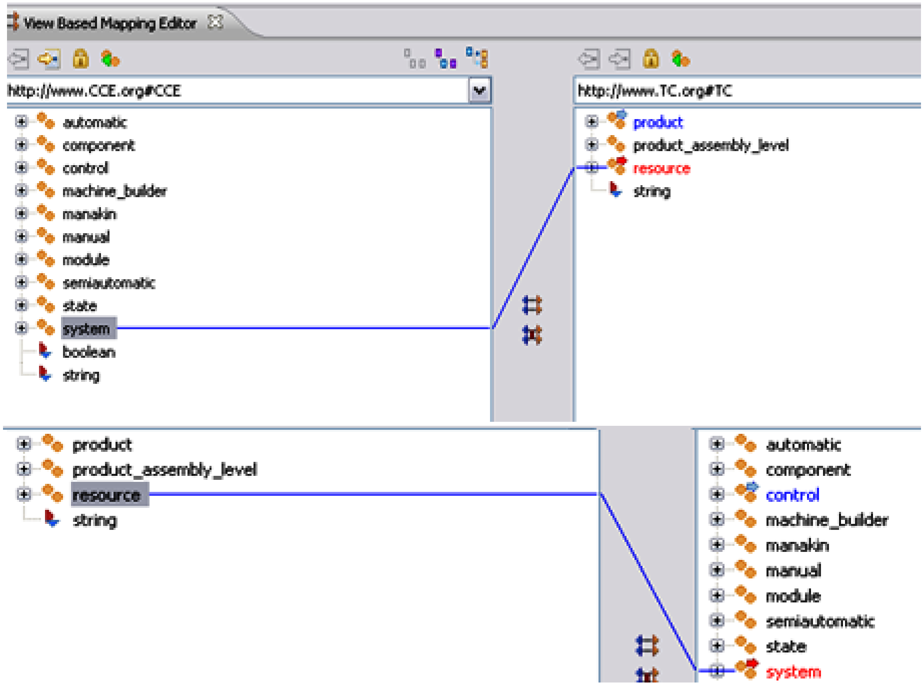
Figure 72: Seamless collaboration through mediation

Similarly, within the same organisation, the need to use information from multiple data sources that have been developed separately, with conflicts in the structure and semantics of these disparate data sources, create major obstacles. For example, in PD, the resource is called ‘station’ while in the CCE tool it is called ‘system’, hence station 1900 and system1900 are essentially the same concepts. An effort has been made to minimise such differences especially by using ontological representations of domain knowledge, however, sometimes it is a requirement to keep the terms different to distinguish, for example, between real and virtual resource etc. Therefore, there is a need to run a mediated query so that a query requiring a certain station, let’s say, “station 1900” should also return “system 1900”, if required. Similarly, the concept ‘product’ is defined and treated differently in TC and CCE e.g. in TC, the concept product is an independent entity with features and parts while in CCE tool, it is defined as a non-control type of component. Therefore the data mediation is required when there are multiple versions of truth within one community. Mediation is required for the following mostly encountered cases during the experimental case studies:

- Same term mean different and refer to separate concepts (Product)
- Different terms mean same and refer to same concept (System and Station)

Ontologies formally specify the terminology used to describe functionalities and behaviour. Mediators allow interoperability among heterogeneously described concepts / sub-concepts, a specific category of mediator being the data mediator. The mediator layer is required to help smooth information flow and retrieval. In the run time environment, the query may invoke the mapping files, if required, to return the results of the same concepts from many ontologies. Therefore if the output from all the applications is required, a mediated query can be run and any particular concept may be retrieved from several applications defined differently in disparate applications. Therefore, the mediator can integrate multiple sources harmonizing semantic / syntactic heterogeneity.

Figure 73 shows a two-way mapping file between CCE & TC, created in the WSMT, where resource and system are mapped and mediated.



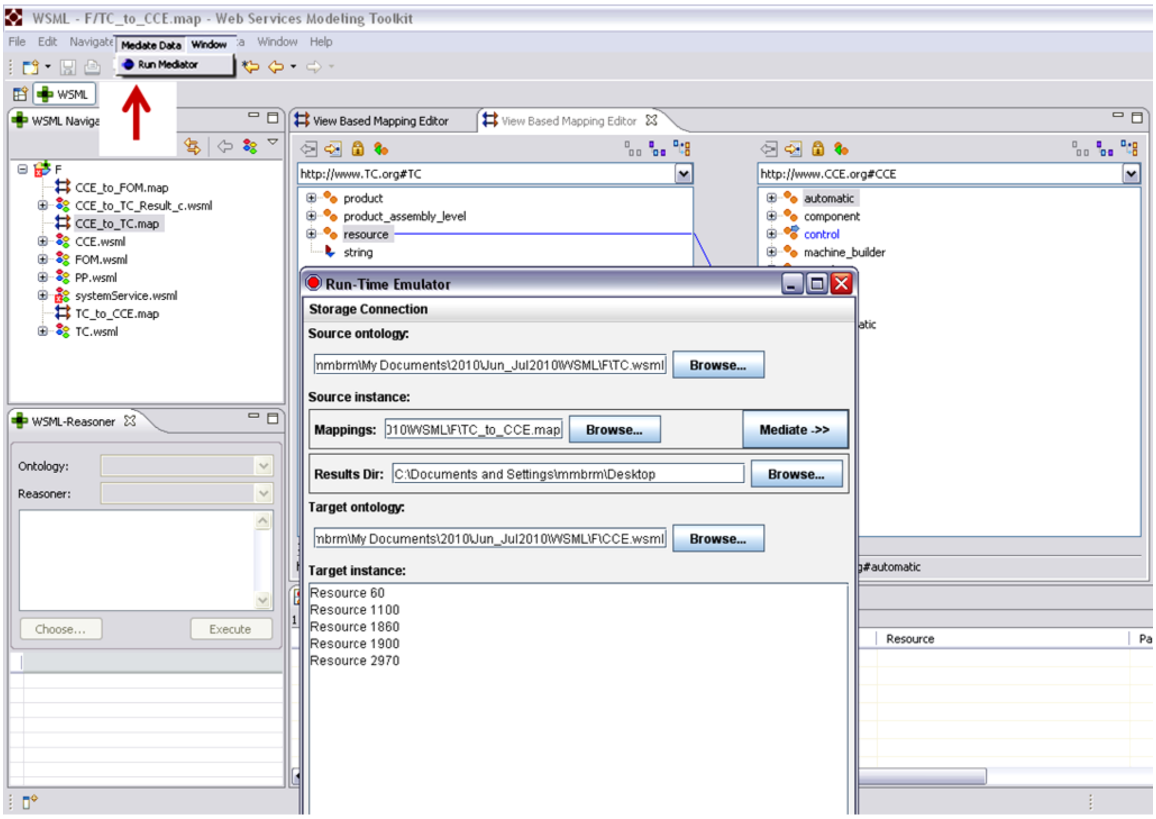
http://www.CCE.org#CCE

http://www.TC.org#TC

Figure 73: Example of a 2-way mapping file between CCE & TC (design time)

In the current (java) implementation, the mediators act as context interchange when data representation conflicts occur. Thus the mediator will transform data, when required or requested, from source to receiver. This can be implemented for concepts or properties of concepts as described in the mediator based COIN strategy (Goh et al, 1999; Bressan et al, 2000).

Figure 74 depicts the usage of the mediator in the run-time environment. The user needs to select the source and target ontologies to get the mediated results.



Run-Time Emulator

Storage Connection

Source ontology: nmbrm\My Documents\2010\Jun_Jul2010\WSML\F\TC.wsml Browse...

Source instance:

Mappings:)10\WSML\F\TC_to_CCE.map Browse... Mediate ->>

Results Dir: C:\Documents and Settings\mmbrm\Desktop Browse...

Target ontology: nbrm\My Documents\2010\Jun_Jul2010\WSML\F\CCE.wsml Browse...

Target instance:

Resource 60
Resource 1100
Resource 1860
Resource 1900
Resource 2970

Figure 74: Run-time mediators in WSMT editor

The mediator can be run specifically to check terminology differences across the enterprises. The data mediation approach uses source-target ontologies to translate between, which may change their roles depending upon which view and context of data is required. The structure and semantics of source and target ontologies wishing to share information may differ hence the data mediator will translate structure and semantics of the ontologies exchanging information.

6.8.2. Support Activity 2: Library Characteristics

The machines and components declared to be under the *reconfigure* and/or *redesign* category need to be modelled and simulated for virtual verification. The reuse of knowledge for the reconfiguring / redesigning process is again vital for rapid and confident decision making. The KB system will help in introducing characteristics of each component of the library and provide an easy way to search and select the requisite component.

6.8.2.1 Summary of the Support Activity 2

Table 21 summarises the support activity 2 provided in the ALDIMS.

Support Activity 2: Library Characteristics	
Industrial Scenario	When the domain expert intends to design a new machine OR the KB system categorises a machine / module to be reconfigure / redesign, the user need to select the most appropriate modules / components to design the machine OR corresponding tools required on the machine
Requirement	Efficient search and retrieval of required tools Efficient search and retrieval of required modules / components
'As-Is' at Ford	Selecting appropriate tools / designing new machines may consume hours or days, consulting catalogues and studying technical guidelines and specifications.
'To-Be' Approach	Efficient search and retrieval of the best component/module of the machine against stated product requirements When a component is selected from library, it must add rules so that user can find whether the selected component is right for the purpose
How	To provide annotated detailed properties linked to the machines / modules / components of machines. This is achieved by attaching additional information that describes physical characteristics and functional details of library of components / modules of the machines This enables to check whether a component with relevant characteristics is already available to be reused, or falls within the scope and category of the investigated component so that requisite modifications can be carried out.
Demonstration	Construction of explicit machine characteristics Selecting best available resource against stated requirement

Table 21: Summary of support activity 2

The machines, its modules and components, are defined in terms of their functionality. The assembly operations are decomposed into atomic process steps as well as the machines are decomposed into basic independent functional units. In this way, products, processes and resources are converted into basic building blocks and the "need-technique-skill" approach is achieved, which corresponds to features, methods and capabilities of the products, processes and resources. In simplistic terms, the user presents the 'need' to be fulfilled, the BoP provides the 'technique' to fulfil the need while the resource provides the 'skill' to accomplish the technique as shown in Figure 75.

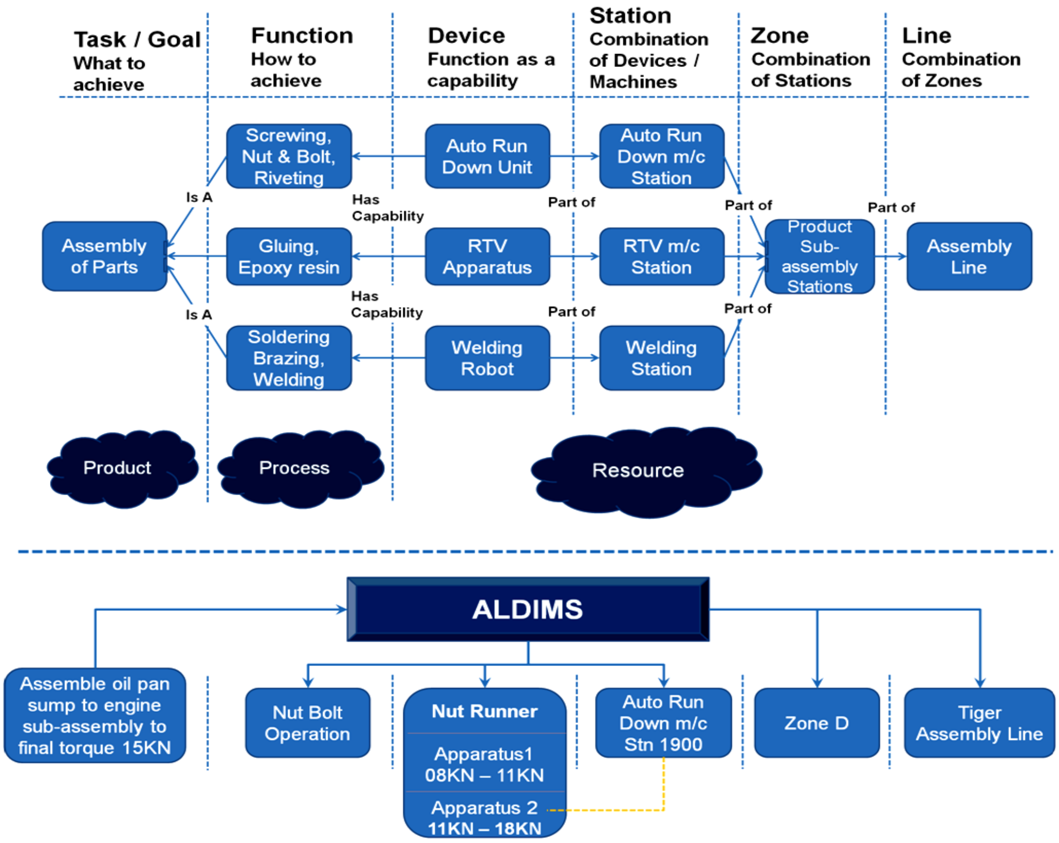


Figure 75: The 'Need-Technique-Skill' approach for selection of equipment

This research approach is a mixture of the concepts described by Kitamura et al (2006) and Jarvenpaa et al. (2011). Kitamura's approach defines devices in terms of 'function-behaviour', whereas Jarvenpaa et al (2011) has described 'capability taxonomy' of the machines. The current approach is an amalgamation of the two approaches as it defines taxonomy of the functions of the resources and links it to the BoP steps which in turn are linked to the BoM.

When a component is selected from the library, the requisite properties and rules can be added so that the user can find whether the selected component is right for the changed product. The component which is being selected was good for the previous program but is it really useful for the new program (changed product) and can it actually perform the task for which it is being selected? When the new machine or machine's components are being selected, it is imperative to know its ultimate capacities as it would virtually perform the intended function, would it operate in real world as intended?

Chau (2007) pointed out that engineers have to spend considerable effort in searching the knowledge they need whereas Allee (2000) and Roy (2007) claims that experience is, in some way, always perishable. Hence an effort is made here to address these concerns so that experience can be preserved by transforming it into reusable knowledge as well as search and retrieval time for the required information can be reduced.

A similar challenge exists at Ford, for example, the CCE tool has been developed with the aim of creating a 'Gold Standard Library' - (GSL) of modules and components of the decomposed machines. It is intended to attach additional information that describes physical characteristics and functional details of the library of components. When this information is added and attached to the models of the modules and components, it enables engineers to query the library that performs a specific function or fulfils a certain criteria e.g. weight bearing capacity. Therefore, the inherent capabilities of the machine/modules/components can be made available readily without going into cumbersome details of simulations and carrying out hefty calculations. This enables experienced engineers to make the selection process efficient while offering an intuitive guide for the novice engineers.

The intended approach is that the process engineer will interrogate the knowledge base to establish whether a component with relevant characteristics is already available to be modified or falls within the scope and category of the investigated component. Thus the experience of specialists is transformed into an empirical rule-base and the process is systemised, formalised and transformed from judgemental adjustment to confident decision making. An example of a library component and associated properties added in the KB system is, for example, from the 'Oil Pan Run Down' machine on the Tiger Assembly Line at Dagenham Plant as shown in Figure 76. The CLR module is an independent unit on the Oil Pan Run Down machine and its components, i.e. Clamping Unit and Rotary Plate, with the function being grip, translate and rotate, are considered for the explanation of the concept.

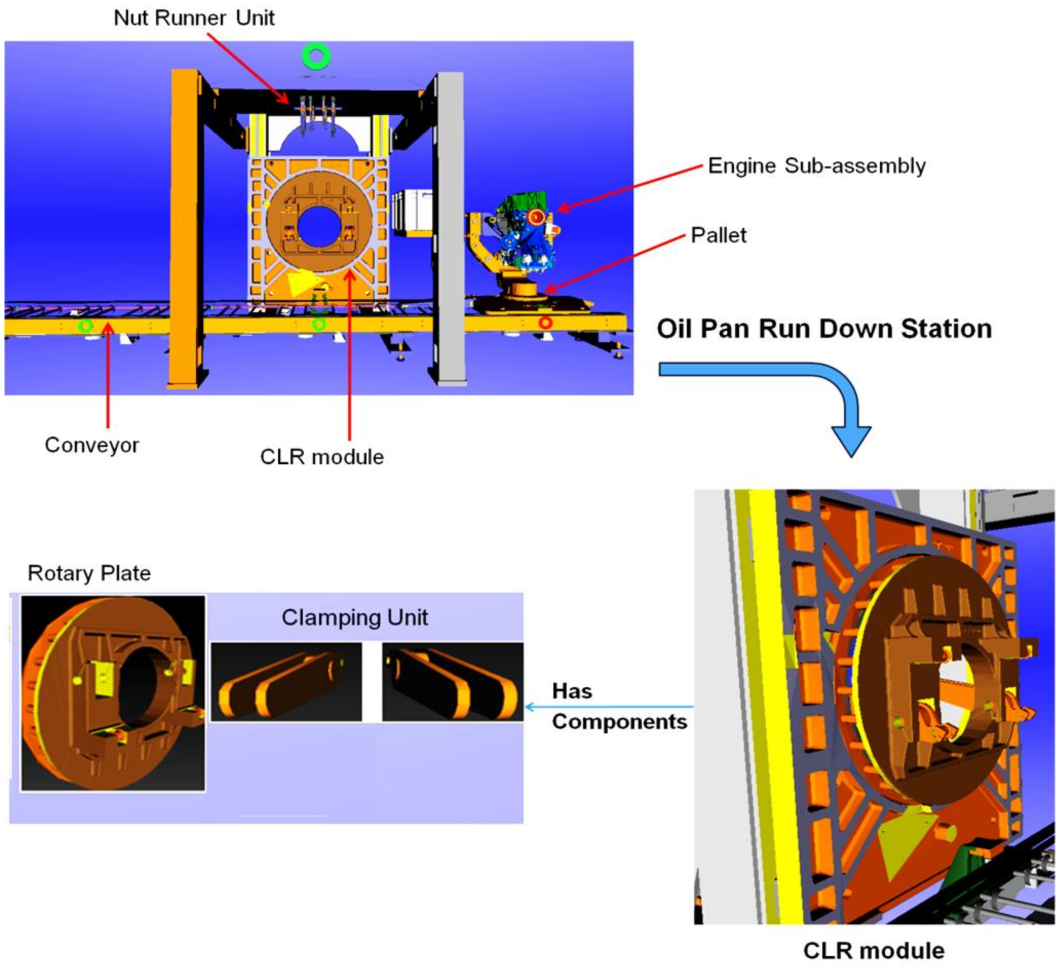


Figure 76: The CLR module used on the oil pan run down machine at Ford

The IMS+ System has a CLR module having properties e.g. payload of ‘X’ kg, raise distance ‘Y’ m, speed ‘Z’ m/s, etc. This system and its module may be used for other engine programs on same / different assembly line. It is required to carry out complete assessment of machines / components before actually selecting / using them. For example, if Sigma engine is using machines of Fox engine, it is necessary to know functional and inherent capabilities of machines / components before selecting / using them. The stated CLR module is part of library of modules being developed by the CCE tool after the BDA project. The developed KB system can help to evaluate characteristics / capabilities of the selected modules / components. For example, the component which is being selected for a new engine program, was meant for the previous program, but is it really useful for the changed product in the new program and can it actually perform the task for which it is being selected. When a component is selected from the library, it must add context properties, capabilities and rules so that the user can find whether the selected component is right for the new product assembly. Likewise which component performs which process step and how the machine component is linked to the parameters of the process? When the new machine or machine's components are being selected, it is imperative to know its ultimate capacities as it would virtually perform the intended function, would it operate in real world as intended?

It is recognised that as the PPR are intertwined, therefore, the connection constraints of the three entities are entwined in the current extended approach in the KB system. Each '*Process*' (and its steps) is directly linked to '*Product*' (and its parts) through '*Makes*' the 'Product_Set' and each '*Station*' (and its modules/components) is directly linked to '*Process*' (and its steps) through '*Performs*' the 'Process' attributes. The system establishes relational dependencies and finally rules and axioms to help assembly design / reconfiguration activity.

The first step towards creating a knowledge-base of library of verified machine mechanisms is capturing all the relevant knowledge that is directly related to the usage of a certain equipment. This knowledge includes context, inputs, outputs, domain, scope, fidelity, complexity, accuracy, inherent properties, relations to other entities etc. An example of a CLR module is described below in Figure 77. The knowledge based system has defined properties of the module in terms of ontology and related the properties with functionality and skills of machines. Once the CLR module is translated into ontology, it is no more a standalone equipment now, it is a part of a system (parent class), it has smaller functional units called components (child class), it is related to product (relational association through attribute) and it has the capability of carrying out certain assembly process steps (attribute of class). Each module / component is part of a system i.e. super-class and is responsible for carrying out the overall intended assembly process. Similarly the relations and attributes can be further refined by defining rules and axioms.

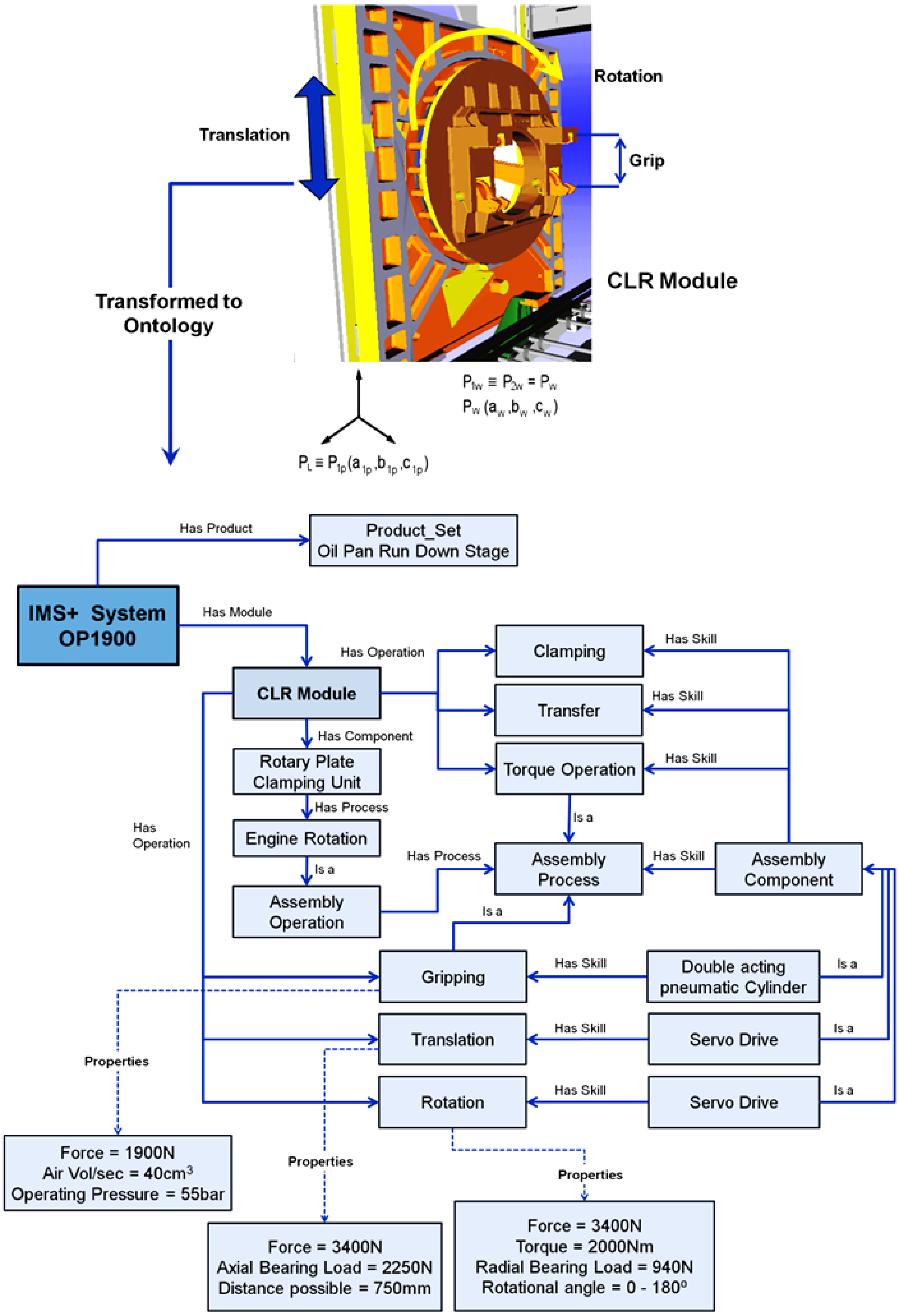


Figure 77: Ontological representation of CLR module

The rules may be added to ascertain how much distance it can travel in z-axis and how much torque is needed. For example, for the CLR module on the oil pan run down station, the possible torque is 2000Nm maximum and the linear distance possible is 750mm, picking capacity as

50kg; placing capacity 1.2meter radius; usage in the OP1100, OP1600, OP 2250 etc. These functional characteristics are defined as ontological properties of the concepts of module / component so that when the respective equipment is retrieved, it can also be analysed for fulfilment of the purpose. Considering the current case study, the engine changes from three-cylinder to four-cylinder, the weight of the engine is increased, there is no explicit knowledge available about the lifting capacity of the CLR module. The rotary unit may lift a heavy engine during simulation yet create problems in real world scenario. For the engine fit analysis case study, the CLR module can be used for the changed engine as the weight bearing capacity is within the range of the gripper. Thus it would help the assembly design activity and selection of proper components, modules and systems accurately based upon the factual knowledge derived from annotated properties, rules and axioms. The system can be used to take into account such attributes of the product with respect to the system limitations, hence, knowledge-driven assembly line design / reconfiguration can be carried out.

6.8.3. Support Activity 3: Hierarchical Classification

The main case study and the first two support activities constitute knowledge capture, document, elicitation, application and reuse while the third support activity deals with knowledge organisation. Knowledge organisation demands a structured hierarchy of concepts for better visualisation and analysis.

6.8.3.1 Summary of the Support Activity 3

Table 22 summarises the support activity 3 provided in the ALDIMS.

Support Activity 3: Hierarchical Organisation	
Industrial Requirement	**Description**
Taxonomical arrangement of PPR entities	Categorisation of knowledge helps logical structuring, indexing, faster search
Knowledge organisation and visualisation	Organisation of automation resources and process steps in hierarchical order as well as retrospective analysis of building a line from smaller modules / components Hierarchical classification helps understand and visualise the complex taxonomy of the assembly line as well as an organised presentation of complex information

Table 22: Summary of support activity 3

An ordered arrangement and classification of the domain entities is also achieved in the developed KB system. The use of ontology and a correctly defined conceptual model of the domain produce automatic taxonomy of the concepts thus helps understand the structure of the domain. In ontology, ‘Is-A’ and ‘Part-Of’ relations create hierarchy of the concepts / properties. A snapshot of the hierarchical order in the ALDIMS tool is shown in Figure 78:

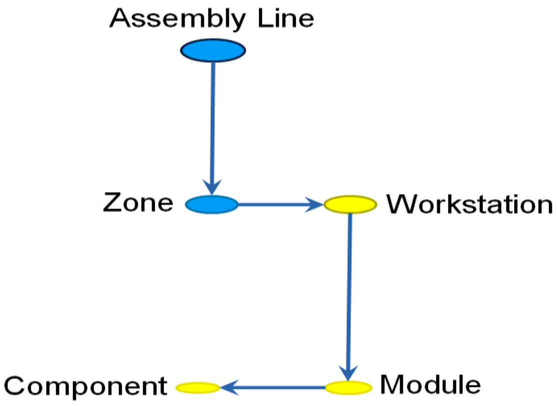
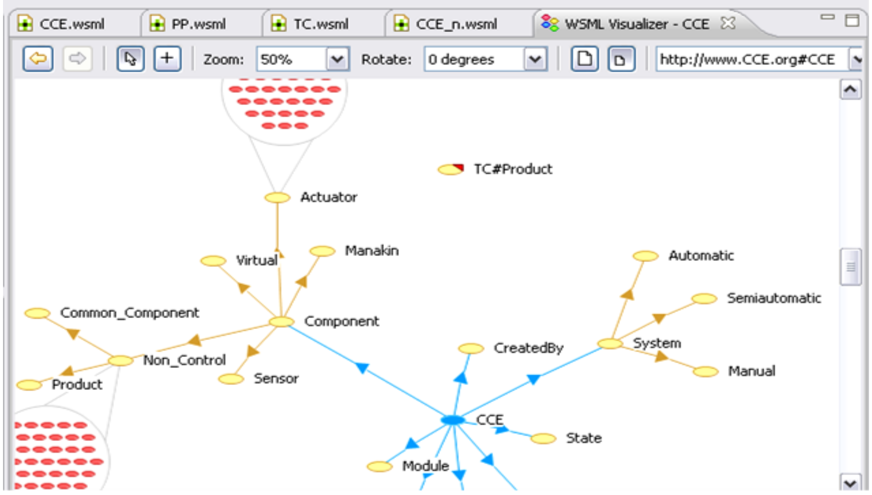


Figure 78: A snapshot of the taxonomical classification of assembly line in ALDIMS

The system established the hierarchy of requirements, functions and capabilities for products, processes and resources. The arrangement of concepts in hierarchy in the ALDIMS is shown in Figure 79:



Zoom: 50%
Rotate: 0 degrees
http://www.CCE.org#CCE

Figure 79: A snapshot of the taxonomical classification of Resource (CCE) ontology

Thus the information classification based upon reasoning capabilities is achieved through a basic standardised framework. The developed KB system produces functional and structural hierarchy, as a result, this constitutes another virtual hierarchy i.e. assemblability of the engine on the assembly line. The developed system adopts hierarchical assemblability approach i.e. define the assembly planning as a hierarchy of requirements, functions and capabilities. Ontology can be used for defining agreed vocabulary and cataloguing to highly organised knowledge bases with intricate rules and axioms. The goal of the KB system is to gather all pertinent information and convert it to actionable knowledge. This knowledge can be arranged into product and resource hierarchies and process taxonomies. Currently there is no hierarchical classification in the prevalent systems to support the hierarchy defined in CCE tool. The ontology of the CCE has been organised in taxonomical form as described below:

concept **System**
concept Automatic subConceptOf System
concept Semiautomatic subConceptOf System
concept Manual subConceptOf System

Controlled vocabulary through ontology adds conciseness and understanding while taxonomy brings organisation. As a matter of fact the hierarchy is automatically generated depending upon the membership of individuals and the logical constraints imposed upon through properties as shown in Figure 80.

```
System
+------- Automatic
|        +------ Comau
|        |       +------ Op60 System
|        |       +------ Op500 System
|        |       +------ Op1100 System
|        +------ Krause
|        |       +------ Op1860 System
|        |       +------ Op1900 System
|        |       +------ Op2970 System
|        +------ ABB Robot
|                +------ Op1560 System
|                +------ Op1600 System
|                +------ Op2800 System
+------- Manual
|        +------ Ford Dagenham
|        |       +------ Operator A
|        |       +------ Operator B
|        +------ Ford Bridgend
|                +------ Operator D
|                +------ Operator E
+ ----- Semi-Automatic
         +------ Comau
         |       +------ Op1260 System
         |       +------ Op1300 System
         |       +------ Op1400 System
         +------ Krause
                 +------ Op1580 System
                 +------ Op1650 System
                 |       + ------Module
                 |                +-------Lifting Module
                 |                |       +-------Component
                 |                |                +-------Sensor
                 |                |                +-------Actuator
                 |                |                +-------Non-Control
                 |                |                +-------Virtual
                 |                +-------Engine Off-load Module
                 +------ Op2770 System
                 +------ Op2890 System
```

Figure 80: Example of a populated hierarchy from the system / station level

The line builder needs to know the machine function while the machine builder needs to know component function, the hierarchical view organises the line and machines in a taxonomical structure and provides a complete visualisation of the requisite entity / concept to help engineers speed up and follow line/machine building activity as visualisation guides the user.

concept **Component**
concept Sensor subConceptOf Component
concept Actuator subConceptOf Component
concept Non_Control subConceptOf Component
concept Product subConceptOf Non_Control – CCE*
concept **Product** – TC*

Similarly the component hierarchy arranged in the system is depicted in Figure 81.

```
Component
+------- Sensor
|        + ------ Position
|        |        + ------ GC406
|        |        + ------ GC403
|        |        + ------ GC402
|        + ------ Pressure / Motion Detectors
|        |        + ------ SF-OLS1004
|        |        + ------ Bosch No. 0261 230 099
|        |        + ------ LIPS 100 linear position sensor
|        + ------ Proximity Sensor
|                 + ------ RIPS 503 inductive proximity sensor
|                 + ------ Baumer IFRM 04P 15A1L
|                 + ------ Telemecanique XS608B1PAL2
+------- Actuator
|        + ------ Electric
|        |        + ------ Servo AC Motors
|        |        |        + ------- McLennan Servo Motor 4000rpm 12V 6W 20mm dia
|        |        |        + ------- Linear Servo Motor 530M4T 20 B3, 1kw
|        |        |        + ------- 316R Series Angular Travel Electric Actuator
|        |        + ------ Stepper DC Motors
|        |                 + ------- RS Stepper Motor 1.8º
|        |                 + ------- SMC MXS16-30B
|        |                 + ------- Torchmate Bipolar Stepper Motor, 300rpm
|        + ------ Pneumatic
|        |        + ------ RC 200 Pneumatic Actuator
|        |        + ------ Y4015-50 Pneumatic Actuator
|        |        + ------ Skilmatic EH Pro Pneumatic Actuator
|        + ------ Hydraulic
|                 + ------ Single speed RS P391
|                 + ------ SMC 9209
|                 + ------ 2-Stage Telescopic Actuator
+------- Non-Control
|        +-------Pallet, Table
|        +-------Guarding, Frames
|        +-------Engine, sub-assembly, nut, bolts etc.
+------- Virtual
         + ------ Timer
         + ------ Counter
```

Figure 81: Example of a populated hierarchy from the component level

Similarly assembly process taxonomy standardisation is achieved and is described below for the oil pan run down assembly process on the station OP1900, as shown in Figure 82:

```
Assembly Process (Oil Pan Run Down)
+------- Entry
|        +------ Pallet entry into the workstation
+------- Process Steps
|        +------ Clamp engine plate
|        +------ Raise engine plate to nut running position
|        +------ Carry out nut running
|        +------ Roll over the engine at roll over height
|        +------ Unclamp engine plate
+------- Exit
         +------ Move the engine to the next station
```

Figure 82: Example hierarchy of an assembly process

Knowledge organisation, as described by Sayers (1995), is: "not only the general grouping of things for location or identification purposes; it is also their arrangement in some sort of logical order so that the relationship of the things may be ascertained." This logical arrangement is provided through the taxonomical hierarchy in the developed system. Multiple hierarchies of concepts are designed and developed for optimum organisation, visualisation and reuse of the knowledge. The ALDIMS tool helps to organise the intermingled information thus, classify, catalogue and arrange the available knowledge into hierarchical order.

6.8.4. *Summary of the Case Study and Support Activities*

In assembly line design / reconfiguration scenario, a combination of the functionalities provided through the support activities for engine fit analysis is extremely useful i.e. rapid evaluation through rules / axioms, data mediation, efficient selection of resources and knowledge visualisation through hierarchical classification techniques. The developed system has been successfully test implemented at Ford and the benefits of knowledge based assembly line design have been demonstrated effectively.

6.9 Key Benefits

It is envisaged that the outcome of BDA project in the form of the combined use of CCE tool and the author's ontological KB system (i.e. ALDIMS), will have fundamental advantages in the way future assembly line design / reconfiguration activity is carried out e.g. through reduced lead time, reduced cost and more confident machine design. As the ALDIMS application is applicable in the design of a new production facility as well as retrofitting in an existing line, a generalised combined case is presented in Figure 83.

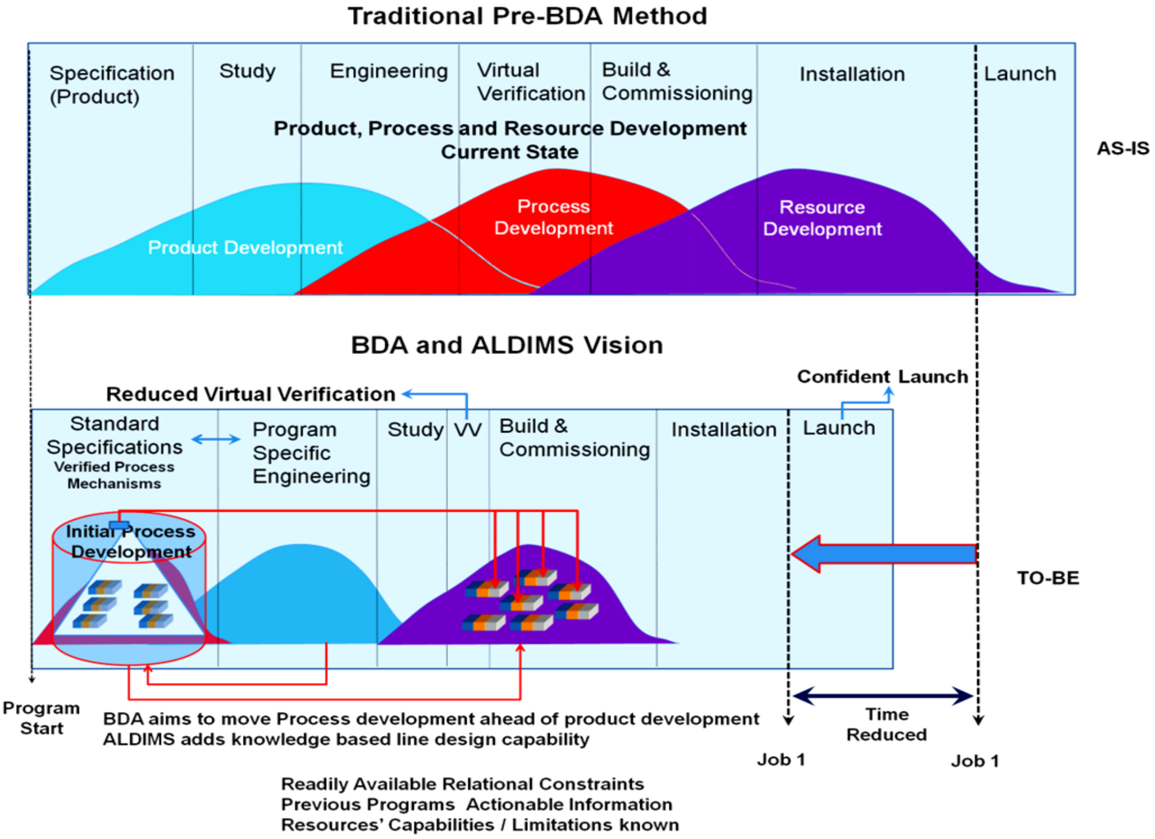

Traditional Pre-BDA Method
Specification (Product)
Study
Engineering
Virtual Verification
Build & Commissioning
Installation
Launch
Product, Process and Resource Development Current State
AS-IS
Product Development
Process Development
Resource Development
BDA and ALDIMS Vision
Confident Launch
Reduced Virtual Verification
Standard Specifications
Verified Process Mechanisms
Program Specific Engineering
Study
VV
Build & Commissioning
Installation
Launch
Initial Process Development
TO-BE
Program Start
BDA aims to move Process development ahead of product development
ALDIMS adds knowledge based line design capability
Time Reduced
Job 1
Job 1
Readily Available Relational Constraints
Previous Programs Actionable Information
Resources' Capabilities / Limitations known

Figure 83: Potential impact and benefits of the ALDIMS

The CCE tool provides verified reusable machine mechanisms, however, to efficiently and effectively utilise these modular resources, a knowledge based infrastructure support is necessary which is provided by the ALDIMS application. The traditional methods followed for new / changed programs is that the product specifications are finalised first, followed by process planning and finally building machines and resources. The BDA project aims to change the existing practices being followed at automotive OEMs especially Ford. In the new approach, the verified reusable modules of machines are available in the form of assembly process operations which can be combined together to form complete assembly machines performing the required assembly operations. Due to readily available machine resources, the time to develop a complete BoP, and in turn, a complete BoR is compressed significantly. The task of process engineers is changed from defining and designing processes to predominantly selecting and verifying processes. Efficient knowledge based support is required to support this new paradigm to select required modules and components effectively and efficiently and this is one of the two major goals of the ALDIMS application (the other one being the rapid engine fit analysis).

In addition to this, due to pressures for the rapid development of new models of vehicles every year, it is not practical to carry out a comprehensive virtual verification of the complete assembly line against new variants of engines. The virtual verification, as described before, is an extremely time consuming process and requires expertise of the domain experts to exploit simulation benefits effectively. This issue is also addressed by the ALDIMS tool. The ALDIMS greatly reduces the level of simulation activity needed by transforming the results of previous simulation (and relevant) activities and events into a reusable knowledge base. The ALDIMS tool provides readily available constraints as well as resource capabilities and limitations to assemble product variants. Therefore, new resources can be built rapidly and reduced reconfiguration time is made possible. As described before, designing a new or reconfiguring an existing assembly line can potentially utilise 75-80% of the existing machines / modules / components. On the average, 40% fall under the category '*use-as-is*' while the same percentage requires slight modifications in the existing modules / components. Therefore, the workstations / modules / components declared to be *use-as-is* or *reprogram* need not to be virtually verified / simulated for assembly hard points. Similarly machines / modules / components declared to be reconfigured may or may not be required to be virtually verified (depends upon the complexity of the module) while machines / modules / components declared to be redesign should be modelled, simulated and virtually verified.

Considering specific practices at Ford, the new program starts with initial product specifications set out by the product designers and released by Product Engineering department (DP4), see Figure 18 in section 3.2.1 and Figure 19 in section 3.2.2. The preliminary study for the new powertrain is carried out by advanced program planning and feasibility (DP5.2) with detailed production strategies and targets and establishing a milestone called program strategy confirmed (PSC). It is envisaged that the DP5.2 can achieve the PSC milestone with greater confidence and in less time after the availability of the ALDIMS tool both for new and changed programs. Machine Builders (DP6) get the targets for mass production through Forward Planning (DP2) and Manufacturing Feasibility (DP3) which is called 1st order to design and build new assembly machines. DP2 and DP3 usually do not have process-resource limitations readily available which now can be provided in the form of ALDIMS tool, hence, the 1st order can be supplied for machine build with initial product-resource constraints kown and evaluated in the PSC and manufacturing feasibility through ALDIMS application (and varified through the CCE tool).

On the other hand, product engineering domain (DP4) can benefit by awareness of process limitations being provided via ALDIMS tool and carry out the constraint based product-resource dependency analyses which is not available in the current commericially available software tools. With the available mapping and relational constriants, communication with suppliers and espeically machine tool builders can be started earlier and rapid development of machine tools and virtual verification of workstation can be achieved to reduce problems in the commissioning and ramp-up stage hence reducing overall lead time for PPR design and development therefore shortening time and cost saving is achieved.

6.10 Summary

This chapter has described the main case study alongwith key facets of the ALDIMS tool to prove the usability and effectiveness of the system. The chapter started with an experimental evaluation of the research concept on the Festo rig which is a small scale replica of the actual assembly systems. This experimental study proved vital before pilot implementation in industrial case. Several issues and limitations were identified and removed accordingly to improve the system. The developed system can be used to efficiently evaluate the effects of product change on existing assembly workstations i.e. engine fit analysis, this analysis is supported by relevant support activites. These support activites include data mediation, efficient selection of resources and hierarchical arrangemenet of domain concepts. The system caters for data mediation issues usually encountered in large scale global organisations. In addition to this, the capability for the knowledge driven selection of resources from library of resource and its decomposed units is discussed. The system's capability in terms of hierarchical classification and organisation has been elaborated and finally anticipated benefits of the developed system are investigated.

CHAPTER 7. ALDIMS EVALUATION

7.1 Introduction

This chapter provides insight into the evaluation processes used to assess the effectiveness of the new approach and the developed system with comparison of the existing approach. It is beyond the scope of this thesis to properly evaluate the capabilities of the ALDIMS tool, however, some evaluation methods are presented to prove the concept and potential to adopt the new approach. The author believes that this research can have both short term and long term effects on the industrial practices especially in the automotive industry. The evaluation of the developed system is divided into three main sections:

(i) Engineering process based evaluation and new resultant business process

(ii) Time, cost and quality based evaluations along with viability of the new approach.

(iii) Evaluation of ALDIMS tool to prove the viability of the new approach

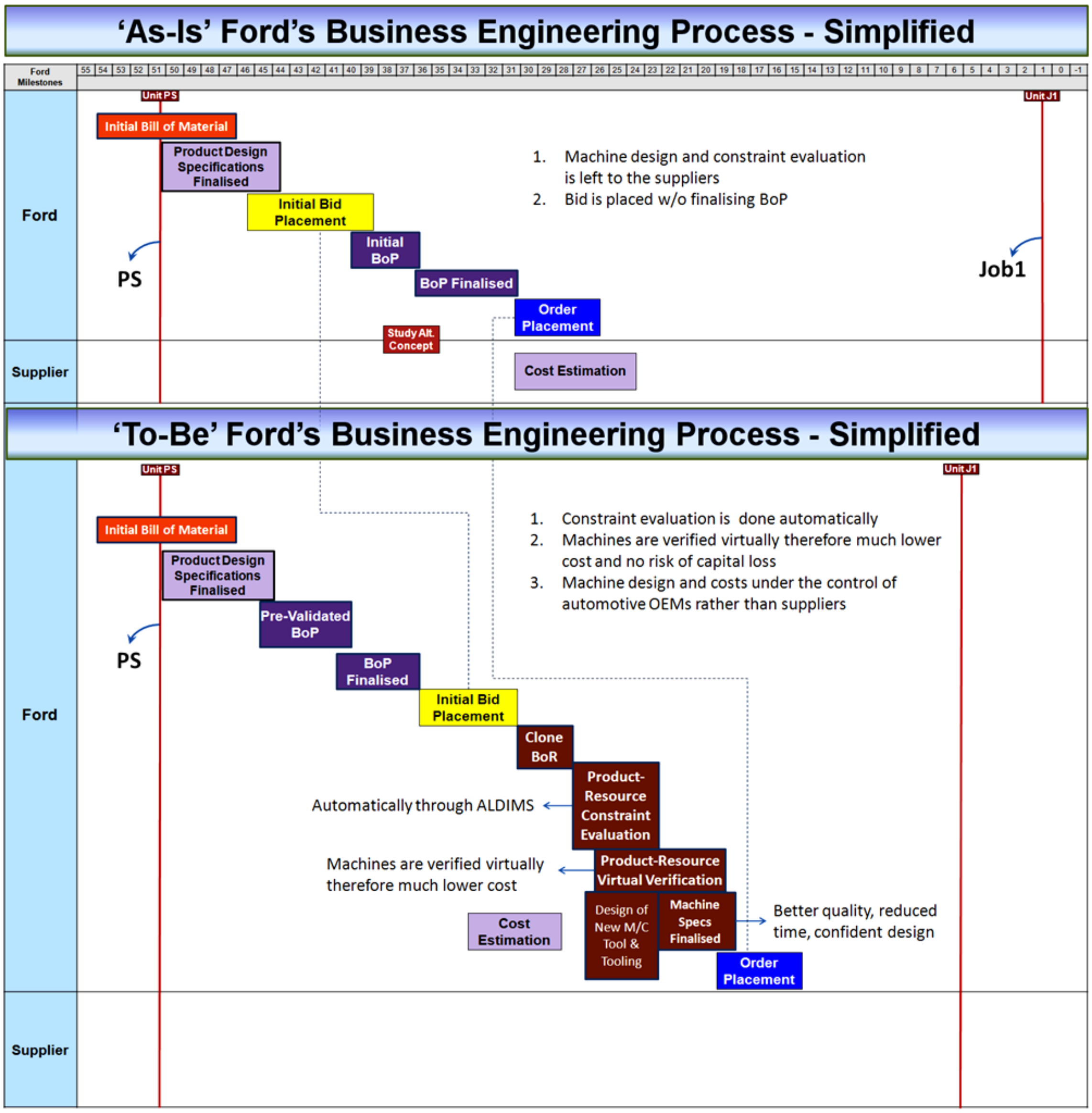
The comparisons and evaluations are made on the supposition that the ALDIMS is in a steady state i.e. the application is fully functional, rule-set is complete, set-up issues resolved and it operates in a stable and consistent manner against user's queries.

7.2 Engineering Process Based Evaluation

It is intended that a comparison of the existing practices against the proposed future state will outline the benefits of using CCE modular tool in conjunction with the KB system and provide a roadmap to increased efficiency and reduced costs. The comparison of the 'As-Is' and 'To-Be' approaches is provided in terms of time, cost and quality, the aim is to reduce lead time and overall cost with the same or increased quality. To better understand the potential advantages of the ontological knowledge base system implementation at Ford, a comparative analysis of the most relevant activities from one of the standardised tasks, done before and after the implementation of the ontological knowledge based is given in the following section.

7.2.1. Impact on Ford's Current GPDS

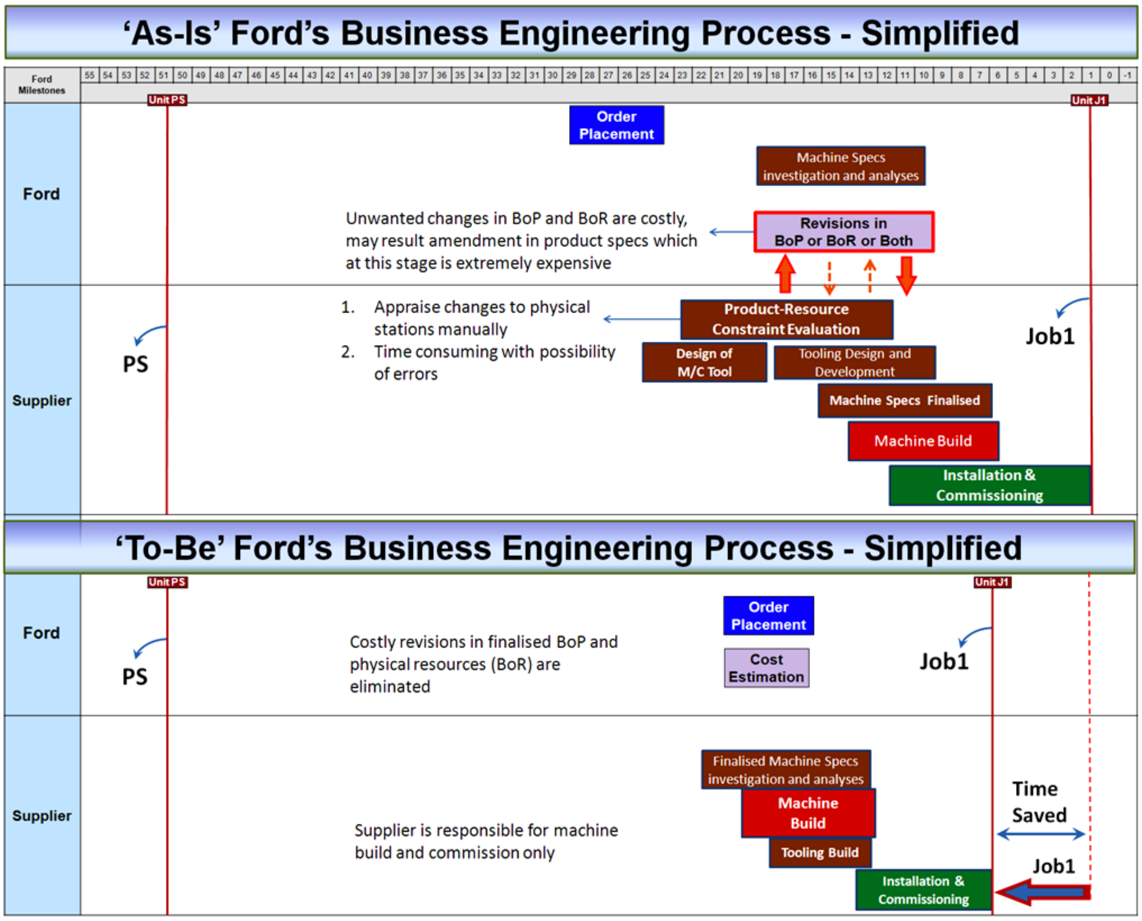
In the Figure 84 and 85, Ford's Global Product Development System (GPDS) is presented in a simplified way. The GPDS consists of standard set of activities and practices to product design / development cycle.



1. Machine design and constraint evaluation is left to the suppliers
2. Bid is placed w/o finalising BoP

1. Constraint evaluation is done automatically
2. Machines are verified virtually therefore much lower cost and no risk of capital loss
3. Machine design and costs under the control of automotive OEMs rather than suppliers

Figure 84: Comparison of ‘As-Is’ vs ‘To-Be’ approaches (1st half)


'As-Is' Ford's Business Engineering Process - Simplified
Ford Milestones
55 54 53 52 51 50 49 48 47 46 45 44 43 42 41 40 39 38 37 36 35 34 33 32 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 -1
Unit PS
Unit J1
Ford
Supplier
Order Placement
Machine Specs investigation and analyses
Unwanted changes in BoP and BoR are costly, may result amendment in product specs which at this stage is extremely expensive
Revisions in BoP or BoR or Both
1. Appraise changes to physical stations manually
2. Time consuming with possibility of errors
Product-Resource Constraint Evaluation
Design of M/C Tool
Tooling Design and Development
Machine Specs Finalised
Machine Build
Installation & Commissioning
PS
Job1
'To-Be' Ford's Business Engineering Process - Simplified
Unit PS
Unit J1
Ford
Supplier
PS
Costly revisions in finalised BoP and physical resources (BoR) are eliminated
Order Placement
Cost Estimation
Job1
Finalised Machine Specs investigation and analyses
Machine Build
Tooling Build
Installation & Commissioning
Supplier is responsible for machine build and commission only
Time Saved
Job1

PS: Program Start **Job1**: First engine of the new program that comes off the assembly line

Figure 85: Comparison of 'As-Is' vs 'To-Be' approaches (2nd half)

Figure 84 and 85 presents a comparison of (a) the current "As-Is" engineering process i.e. the conventional design and build process using conventional CAD from machine design and normal physical machine build and commissioning against (b) the "To-Be" process envisioned using a combination of ALDIMS constraint evaluation followed by virtual verification before physical machine build.

7.2.1.1 Problems Identified with "As-Is" Approach

- Changes required in resources against new / changed product are appraised manually on physical resources
- The order for machine design and build is placed on the machine supplier before BoR design finalisation. The order is placed earlier in anticipation of saving time but it leaves uncertainty and delays often occur in program targets due to late design changes.
- Problems during installation and commissioning phase further delay the program targets.

7.2.1.2 Key Differences in the "To-Be" Approach

- Generation of a generic BoR through 100% virtual verification of all the workstations in order to minimise later revisions on physical resources
- Bid and order are placed after virtual finalisation of BoP and BoR respectively
- Machine design and cost estimation become under the control of automotive OEMs rather than suppliers, this is enabled by virtual engineering
- The constraint evaluation is mostly automatic and in a virtual environment. This (i) significantly reduces the size of the virtual verification task for the automotive OEMs, (ii) significantly reduces the skill level required, therefore, rapid and virtual verification of only those workstations / modules / components is carried out which is essential e.g. in the *redesign* and *reconfigure* categories with ALDIMS.

7.2.1.3 Potential Impacts

In the new suggested approach, the majority of the potential assembly hard points are evaluated automatically or through virtual verification, therefore, considerable reduction in time and, in turn, cost is achieved. The key thing is that the changes required in the resources are appraised virtually, therefore, substantial reduction in cost is possible compared to the current practice of change appraisal in the physical resources. The machine build is under the control of the end-user, therefore, realistic program targets and justifiable costs can be accomplished. Also the cost of assembly lines will successively reduce for changes / new programs as the issues encountered can be documented as lessons learnt and can be avoided for the next programs.

The revision cycles in the BoP and BoR cause significant time losses and can also be detrimental especially if these, consecutively, affect the finalised product specifications. These recurring cycles are hard to quantify explicitly in terms of time, cost and quality due to the uncertainty in their occurrences and outcomes. The ALDIMS tool will assist rapidly evaluate potential changes i.e. 'Δ' required in the existing line, therefore, Δ is the difference between new requirements and the existing solutions. Thus the end user has more control over the machine design and cost analysis activities.

7.2.1.4 Discussion and Analysis – 'As-Is' and 'To-Be' Approaches

The Figure 83 and 84 provide a simplified comparison between existing and suggested approaches especially highlighting the CCE and ALDIMS tools usage and advantages. The three major changes at the automotive OEMs in the current activities based on the achievements during the successful BDA project are suggested. The bid placement is made after finalisation of the

BoP; the order placement is made once the virtual verification of the new / changed machines is complete; and finally the BoP and BoR recurring design verification revisions and cycles of changes between automotive OEMs and the machine suppliers are eliminated by (i) automatic evaluation of product-resource constraints through ALDIMS tool and (ii) rapid cloning of new resources through CCE tool. These changes will ensure a confident build of machines by the machine suppliers and help improve product launch timing.

The comparison shows reduction in lead-time in the design / reconfiguration of the engine assembly line process. The ALDIMS and CCE tools help to clone the existing / generic line and virtually verify the resources which need redesign / reconfiguration. The bid placement milestone starts later, similarly the order placement is also made later. The late start of the machine build activity is hugely compensated by the automatic evaluation of constraints and rapid virtual verification activity, thus, the overall lead time is reduced with lower revisions and lower risks of late program approvals and less dependence on the machine tool builders.

The ‘As-Is’ process provides limited opportunity for a knowledge based line design because the current software applications, due to lack in capability to rapidly analyse the product-machine constraints, do not promise a complete engine fit analysis through virtual verification of the line. A limited engine fit with physical machine / components is carried out only at the machine builders. The CCE tool has introduced machine models with virtual engineering tools, therefore, virtual verification of the engine fit analysis is made possible. The aim of the BDA project is to virtually build a complete engine assembly line and clone new lines from the generic line for new engine programs or make amendments in the existing line for variants of existing engines. This creates the need and the opportunity to use knowledge based approach to line design and reconfiguration. Once the virtual models of all the workstations are available, it is aimed to reuse the available knowledge by documenting the simulation results in a reusable form.

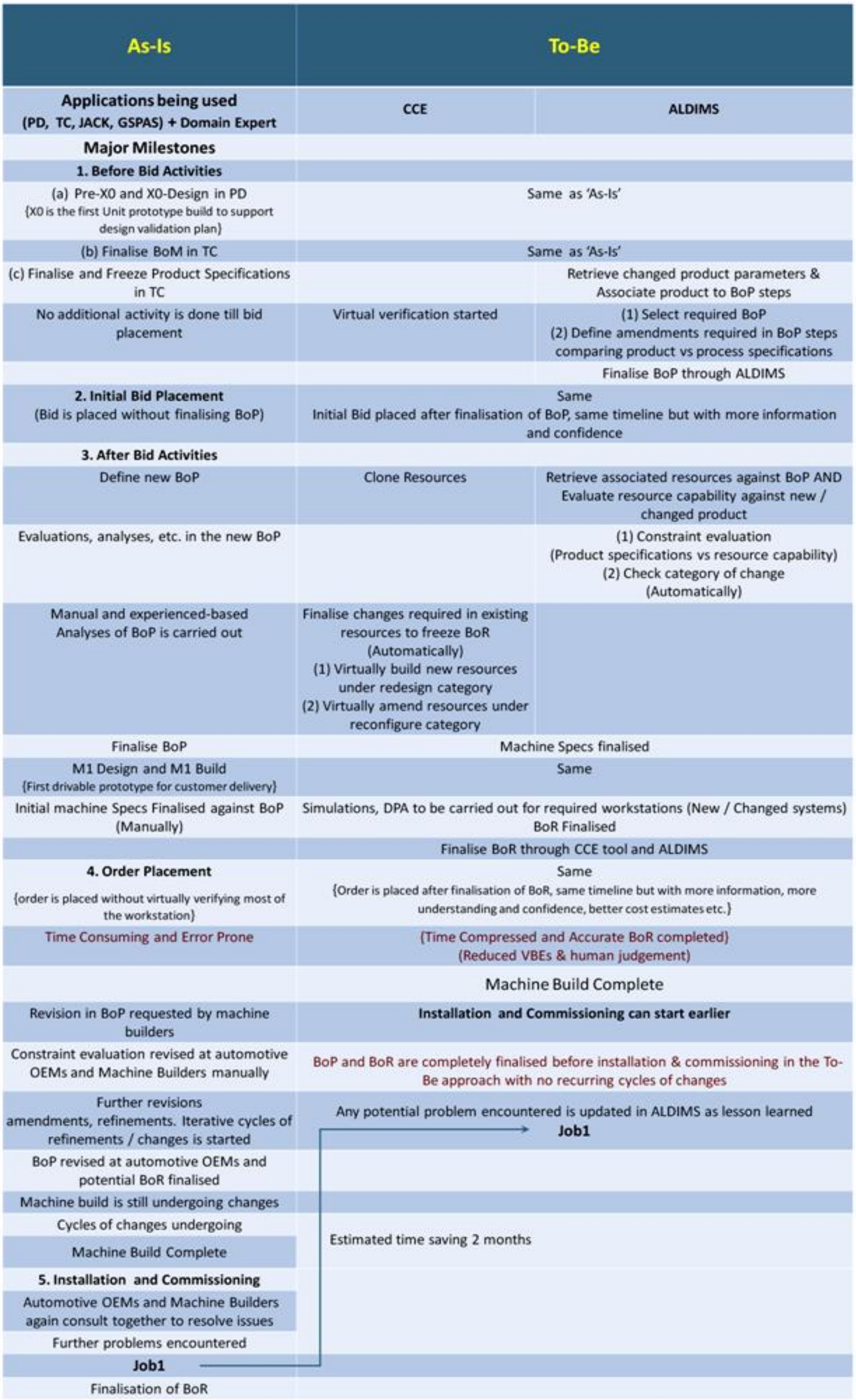
The virtual verification process through simulation of the 3D models is translated into parameterised constraint analysis, therefore, for any change in the existing program, a quick retrofit in the existing machines is ascertained instead of repeating the virtual verification process. Similarly, for a new engine program, the existing machines may be cloned and changes required on the cloned machines can be consulted through the knowledge based system and implemented in the CCE tool.

The ALDIMS tool will enable quick analysis and earlier resolution of assembly hard points, fixture clashes, tool access and promote knowledge reuse with optimal development of assembly machines. The test implementation results showed that the machine feasibility issues are resolved quickly, product assemblability is achieved earlier, error proofing is provided through rules, axioms and inter-dependency relations among product specification, process planning and resource designing and early confirmation, that product specifications are achievable or not, is achieved. Another advantage is more control of the cost estimation and, in turn, budget organisation process from vendors to the automotive OEMs. The automotive OEMs need to place order for the changes in the existing line rather than the complete line order.

In this way, the automotive OEMs can take control of the line design, build as well as cost analysis process. Thus the BDA project has helped the Ford Company to take control of the machine design and cost analysis activities while the suppliers should be responsible for machine build only. Currently machine design and build is almost solely in the hands of supply partners. The underlying difference is that the machine design activity is shifted from supplier to the automotive OEMs and, in turn, virtual verification of the machine-product constraints is done automatically on the virtual models of the product-resource (rather than on the physical resources) through knowledge based system.

7.2.2. Milestone Basis Comparison

The Table 23 provides milestone basis comparison and shift of activities after BDA project outcome. The complexity of the activities is again simplified to get an understanding of major activities and milestones along the new program / changes in program for the PPR facets. It provides a simplified comparison between existing and suggested approaches especially highlighting the CCE and ALDIMS usage and advantages. The 03 major changes in the current approach are suggested as CCE tool and ALDIMS get initial ongoing implementation. The bid placement is made after finalisation of BoP; Initial finalisation of machine specifications is replaced with finalisation of machine design with the help of CCE and ALDIMS; and finally the order placement is made once the virtual verification of the new / changed machines is complete. These changes in activities / change in sequence of activities transform the line design / reconfiguration to a knowledge based design and reconfiguration and will ensure a confident build of machines by the machine suppliers avoiding recurring cycles of changes required in BoP and/or BoR along with significant reduction in the overall lead time from PS to Job1.



As-Is	To-Be	
Applications being used **(PD, TC, JACK, GSPAS) + Domain Expert**	**CCE**	**ALDIMS**
Major Milestones		
1. Before Bid Activities		
(a) Pre-X0 and X0-Design in PD {X0 is the first Unit prototype build to support design validation plan}	Same as 'As-Is'	
(b) Finalise BoM in TC	Same as 'As-Is'	
(c) Finalise and Freeze Product Specifications in TC		Retrieve changed product parameters & Associate product to BoP steps
No additional activity is done till bid placement	Virtual verification started	(1) Select required BoP (2) Define amendments required in BoP steps comparing product vs process specifications
		Finalise BoP through ALDIMS
2. Initial Bid Placement (Bid is placed without finalising BoP)	Same Initial Bid placed after finalisation of BoP, same timeline but with more information and confidence	
3. After Bid Activities		
Define new BoP	Clone Resources	Retrieve associated resources against BoP AND Evaluate resource capability against new / changed product
Evaluations, analyses, etc. in the new BoP		(1) Constraint evaluation (Product specifications vs resource capability) (2) Check category of change (Automatically)
Manual and experienced-based Analyses of BoP is carried out	Finalise changes required in existing resources to freeze BoR (Automatically) (1) Virtually build new resources under redesign category (2) Virtually amend resources under reconfigure category	
Finalise BoP	Machine Specs finalised	
M1 Design and M1 Build {First drivable prototype for customer delivery}	Same	
Initial machine Specs Finalised against BoP (Manually)	Simulations, DPA to be carried out for required workstations (New / Changed systems) BoR Finalised	
	Finalise BoR through CCE tool and ALDIMS	
4. Order Placement {order is placed without virtually verifying most of the workstation}	Same {Order is placed after finalisation of BoR, same timeline but with more information, more understanding and confidence, better cost estimates etc.}	
Time Consuming and Error Prone	(Time Compressed and Accurate BoR completed) (Reduced VBEs & human judgement)	
	Machine Build Complete	
Revision in BoP requested by machine builders	**Installation and Commissioning can start earlier**	
Constraint evaluation revised at automotive OEMs and Machine Builders manually	BoP and BoR are completely finalised before installation & commissioning in the To-Be approach with no recurring cycles of changes	
Further revisions amendments, refinements. Iterative cycles of refinements / changes is started	Any potential problem encountered is updated in ALDIMS as lesson learned **Job1**	
BoP revised at automotive OEMs and potential BoR finalised		
Machine build is still undergoing changes		
Cycles of changes undergoing	Estimated time saving 2 months	
Machine Build Complete		
5. Installation and Commissioning		
Automotive OEMs and Machine Builders again consult together to resolve issues		
Further problems encountered		
Job1		
Finalisation of BoR		

Table 23: Simplified major milestones basis comparison of 'As-Is' and 'To-Be' approaches

7.2.3. Review of Changes in Practices / Approach

It is envisaged that the BDA project and the current research will bring about essential enhancements in the current approach of assembly line design / reconfiguration activity. Table 24 summarises changes in existing approach and practices:

Summary of Revisions in Approach and Practices		
Current Practices	**BDA Project Outcome**	
	CCE Tool impacts	**ALDIMS impacts**
Limited Simulation + DPA + VBE + SBE + manual evaluation on physical resources + Tacit Knowledge	Cloning of resources + Simulation	Relational constraints + Rules and Axioms; Tacit knowledge to Implicit Knowledge
Resource building from scratch	Resource building from pre-validated modules and components	Reduced simulation + Quick evaluation of PPR constraints
Information-based + Model-based approach	Information-based + Modular-based approach	Knowledge-based + Constraint-based approach
Reuse previous information	Reuse previous module	Reuse previous knowledge
Issues / Decisions not documented	-	Lessons learnt, issues / decisions documented
Information-based line design	Component-based line design	Knowledge-based line design
Cost estimation on human judgement	-	Quick insight into the realistic cost estimates
Conventional product-process-resource development	Novel approach – Process available before product development	Complementing new approach by efficiently selecting appropriate process modules
Interoperability issues	-	Reduced Interoperability issues
Experience cloning	Resource cloning	Organisational memory, expertise and knowledge cloning

Table 24: Comparison of 'As-Is' and 'To-Be' approaches

7.2.4. Summary of Potential Engineering Uses

The developed system is a prototype to prove the concept in real industrial scenario and provides an effective and efficient workbench for every engineer's desktop with efficient searching, analyses and reasoning capabilities both for experienced and unskilled users. The potential uses of the ALDIMS tool along different lifecycle phases and departments for a few most relevant activities during assembly line design / reconfiguration activity are summarised in Table 25. Assembly line design and reconfiguration requires several activities e.g. PPR mapping and constraint evaluation, decision support, simulation based virtual verification. These activities are exercised during different lifecycle phases of the engine assembly to fulfil specific engineering requirement. The Table 25 provides an overview of these activities from Ford's perspective and the support provided by the ALDIMS tool against these activities.

Definitive Functionalities Required During Powertrain Assembly Line Lifecycle		Scope & Time of the Activity	Current Engineering Activity Fulfilled	Potential Usage in Lifecycle Phases at Ford *	Support Provided by the ALDIMS
1	PPR Mapping	To associate the related entities Rapid information retrieval	Identify association of PPR	DP5.3	Yes
2	Constraint Evaluation	Evaluation of the effects of product change to existing / generic resources	Machine design	DP5.1, 5.2	Yes, automatic & rapid
3	Rules and Axioms	Evaluation of the affects of product change to resources	Machine design, DPA, VBE, SBE	DP5.4	Yes, 1st time introduced
4	Decision Support in M/c Reconfiguration	Machine design	Manufacturing feasibility, machine design	DP5, DP6	Yes, 1st time introduced
5	Virtual Verification through Simulation	Manufacturing engineering	Detailed machine examination for final release	DP5.4	No, but ensures simulation results are reused
6	Data Mediation	Data interaction among localised applications	Common meaning and understanding of data (Support along several activities)	DP5.1, 5.2, 5.3, 5.4, 5.6	Yes
7	Hierarchical Classification of Machines	Cloning of machines for new programs	Resource library of modules	DP5.4	Yes
8	Cost Analysis	Earlier stages of new business case	Investment cost estimation	DP 5.1, 5.2	No, Can be introduced easily

*Lifecycle stages and activities at Ford as described before in Figure 18 section 3.2.1 and Figure 19 section 3.2.2.

Table 25: Overview of ALDIMS support and usage for engine assembly line design from Ford's Perspective

7.2.5. New Engineering Process and Workflow

First a brief of the current major interactions between automotive OEMs and machine builders are depicted in Figure 86.

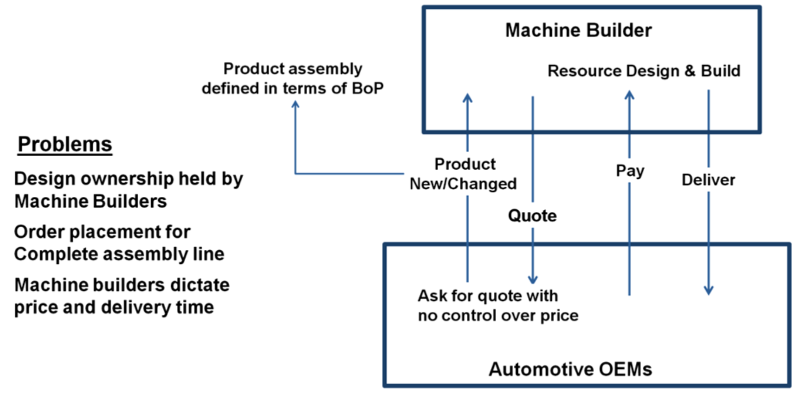

Machine Builder
Resource Design & Build
Product assembly defined in terms of BoP
Problems
Design ownership held by Machine Builders
Order placement for Complete assembly line
Machine builders dictate price and delivery time
Product New/Changed
Quote
Pay
Deliver
Ask for quote with no control over price
Automotive OEMs

Figure 86: Simplified ‘As-Is’ engineering workflow and interactions

Currently, the automotive OEMs send product specifications with initial BoP to get quote of the machines. The machine design, build, manual constraint evaluations and commissioning are all in the control of machine builders, therefore, it increases the overall cost as well as the program targets depend upon pace of the machine builders. With the introduction of faster and modular virtual engineering tool and knowledge based support for the line design, the cost and time aspects come into the control of automotive OEMs.

A review of the changed activities is depicted in Figure 87. Within the automotive OEMs, a new business case in the form of a new product or a product variant from product engineering (DP4 at Ford) is handed over to manufacturing feasibility / manufacturing engineering (DP5.2 and DP5.4 at Ford). The effects of product change can be evaluated with the help of ALDIMS tool with categories of changes explicitly declared in the generic / existing resources. The changes required in the resources will be forwarded to the machine builders for confirmation and design amendments at machine builder’s end. The changed / new designs will be forwarded to the automotive OEMs which will be used for simulation-based virtual verification of constraints and finalisation of build specifications. The machine builders can start building the machines while virtual verification can be carried out at automotive OEMs with the help of CCE and ALDIMS tools. The results of the CCE tool and physical commissioning can be incorporated in the ALDIMS tool as lessons learnt.

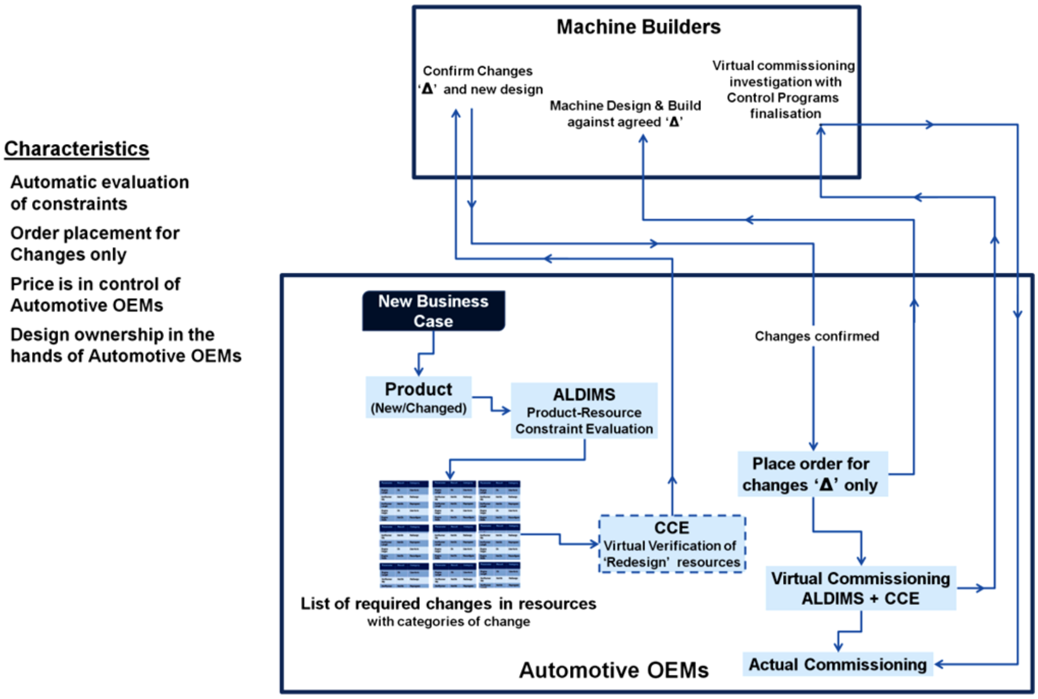

Characteristics
Automatic evaluation of constraints
Order placement for Changes only
Price is in control of Automotive OEMs
Design ownership in the hands of Automotive OEMs
Machine Builders
Confirm Changes 'Δ' and new design
Machine Design & Build against agreed 'Δ'
Virtual commissioning investigation with Control Programs finalisation
New Business Case
Product (New/Changed)
ALDIMS Product-Resource Constraint Evaluation
Changes confirmed
Place order for changes 'Δ' only
CCE Virtual Verification of 'Redesign' resources
List of required changes in resources with categories of change
Virtual Commissioning ALDIMS + CCE
Actual Commissioning
Automotive OEMs

Figure 87: Simplified 'To-Be' engineering workflow and interactions

The new engineering workflow changes:

i. the stages in the process (PPR constraint evaluation stage added at the end user end)

ii. the relationship between the automotive OEMs and the machine suppliers

The new engineering workflow adds constraint evaluation through virtual engineering tool once and then parameterised analysis for subsequent times at the automotive OEMs end.

7.3 Time, Cost and Quality Comparisons

In order to check the viability of the rapid parameterised constraint analysis approach, it is imperative to investigate a comparative analysis of the time and cost saved and potential improvements in quality facet. Based on real engineering data from Ford, informed estimates can be made of the potential time and cost savings and improvement in quality facet of the assembly line design and/or reconfiguration activity that might be possible through the introduction of the ALDIMS tool. The effect of the ALDIMS tool on time, cost and quality of the engineering process of the engine assembly line are described in the next sections.

7.3.1. Time Facet

The BDA project outcome has resulted in virtual engineering support tools for simulation based and parameterised based analysis of the assembly line design, reconfiguration and commissioning activities. The average time required for complete virtual verification of product-resource constraints on a single workstation with the CCE tool is 20 hours[79]. For a complete assembly line of around 150 stations, this time becomes approximately 3000 hrs. This work would be spread out over a number of months depending upon the number of domain experts working with the CCE tool. For example, with the current workforce of 5-6 specialist engineers, it would take around three months at Ford to appraise the changes required to the existing resources against new/ changed product. With the introduction of knowledge based system, these changes can be ascertained a lot quicker e.g. for the entire line, it would require 12.5 hrs for the ALDIMS application to calculate the required changes to the existing resources[79]. Thus the product-resource constraints for the entire assembly line may be calculated in a couple of hours, rather than months, with less experienced engineers working with the ALDIMS tool. The Table 26 provides details of the time savings.

Time Facet		
1	**As-Is**	
	No virtual verification, apparent time saved with no virtual verification later consumes more time for manual verifications on physical resources, changes and amendments	
2	**To-Be – BDA virtual engineering applications support**	
i	**CCE tool only**; Simulation based virtual verification	**Time for Entire Line**
	20 hrs / machine and 150 machine on engine assembly line (*Virtual verification of each workstation before actual build*)	3000 hrs
ii	**ALDIMS only** (parameterised analysis for evaluation of changes required on resources)	
	Evaluation of changes i.e. 'Δ' only: 0.084 hrs / machine & 150 machines *Quick evaluation of product changes in practically no time usually required for variants of engine*	12.6 hrs
	Potential Time savings for appraisal of changes on machines only (For evaluation of Δ, repetition of simulation will have to be exercised in CCE tool)	**99.5%**
iii	**ALDIMS + CCE** (parameterised analysis and reduced simulation-based virtual verifications)	
	Timely evaluation of changes + targeted simulation *Quick evaluation of changes, specific areas to concentrate in simulation analysis and efficiently cloning the available resources for design of new resources usually for new engine programs*	*3000 hrs x 50% = 1500 hrs 1500 + 12.6 = 1512.6 hrs

[79] Results of the validation exercise for ALDIMS in which a comparative analysis of the BDA virtual engineering tools i.e. CCE and ALDIMS were compared with the existing practices

	Minimum Time Saving	**49.6%**

Table 26: Summary of results for times spent 'As-Is' and 'To-Be' approaches

*Current reuse practice for machine design and build = 20-25%
Potential to reuse existing design and build = 75-80%
Minimum savings possible[80] = (75 – 25) = 50%

Thus at least 49.6% of the time can be saved by quick evaluation of changes and then focussed virtual verification through a combined use of the CCE and ALDIMS tools. The time analysis is based upon the Ford's Fox line virtual engineering time comparison exercise carried out during the program planning stages of the assembly line design in the beginning of the program.

Right Decision at the Right Time

The CCE tool aims to carry out virtual verification of the complete assembly line (3000 hrs) and with ALDIMS application, the changes may be evaluated in approximately 12.6 hrs. The reduction in time is crucial to make timely decisions. The current heavy weight software applications can also perform simulation based virtual verification, however, as it takes a huge time, therefore, it is opted to make experienced-based judgemental decisions initially or manual constraint evaluation on physical resources later. The CCE tool is lightweight thus ensures quicker constraints verifications, however, it is still a labour-intensive task and verifying each workstation for frequent product changes is not always feasible. With the CCE tool, it would require approximately three months to ascertain changes, this time span may be too late to provide details of the changes required on the generic line. This issue is directly addressed by the ALDIMS tool. Therefore, the BDA project aims to exercise virtual verification of a complete assembly line once and produce a generic assembly line from the simulation results and any alteration required on the generic line against new / changed product would be calculated as changes required on the generic line. The ALDIMS application can provide the required changes in 12.6 hrs and an early machine design collaboration can be started with a better control of the design process with the automotive OEMs. The reduction in time can have far reaching impacts not only in taking prompt decisions but also controlling the line design activity.

Therefore, it is anticipated that a combined use of ALDIMS and CCE tools can reduce at least 49.6% of the virtual verification time. Formulating rules in the ALDIMS will take, on the

[80] A maximum of 60% is possible {(80-20) = 60%}

average, 750 hours of time of knowledge engineer for a complete assembly line, in its current form. This time will be increased if the total number of rules is increased, it depends on the required quality and usability of the KB system. However, without ALDIMS, the same simulation activity will have to be repeated for every new / changed program.

Typically, there are 28-30 new / changed Powertrain Operation (PTO) programs/year, for example, at Ford. Out of these, there are approximately five to six new programs requiring new build and installation of assembly line, while averagely, 20 to 25 programs require retrofit in the existing assembly line. Out of these, there are six programs with major changes on the engine and eight programs with minor changes on the engine. There are ten programs which require retooling at the shop floor level i.e. the changes in the engine variant are trivial and production engineers accommodate readjustments at the shop floor level, these minor changes have not been taken into consideration while calculating benefits of ALDIMS tool. The extra time spent in defining KCs and formulating rules is well justified when comparing the total number of hours the process engineers / domain experts will eventually spend in repeating the same exercise again and again.

On the basis of this analysis, the author strongly believes that the knowledge based approach will have a significant impact in reducing the overall time spent on the assembly line design activity. It is also admitted that the virtual verification / simulating cannot be completely eliminated nor is this the aim of the research. The two industrial requirements addressed are (i) rapid engine fit analysis especially in the beginning of new / changed programs (ii) to help decide where the expertise of the domain experts needs to be utilised by providing categories of changes required on the existing resources.

7.3.2. Cost Facet

The cost based evaluation is aimed to provide an insight into the KB system support to reduce the cost of developing and building an assembly line. As described in the previous section, the KB system along with modular virtual engineering tool can significantly reduce total lead time, therefore, it is envisaged that it will also reduce the overall cost of the assembly lines. It is believed that the ALDIMS will have significant impact on cost reduction efforts, however, it is also admitted that it is difficult to quantify the cost savings with accurate values. The cost breakdown values are based upon the Ford's Fox line virtual engineering cost estimation exercise carried out during the program planning stages of the assembly line design in 2010 as well as well-established numerical figures of assembly line cost breakdown. The effect of the ALDIMS is analysed in two broader cost reduction categories as described below:

Cost savings in design

Depending upon the business requirement, the ALDIMS tool can significantly reduce time and in turn cost during the design of assembly automation resources. As described before the minimum savings in time with a combined use of CCE and ALDIMS tools is 50%. This is the direct saving possible due to early confirmation of reusable and reconfigurable resources thus avoiding repetition of efforts in designing and building existing resources. The indirect savings in cost could be adopting virtual verification activities results in lieu of prevalent error prone, time consuming and cost incurring activities e.g. manual evaluation of engine fit on physical resources.

If the total changes required in the existing / generic line are represented by 'Δ', then,

Δ = Changes Required
Δ = New Requirements – Existing Solutions
Cost = Δ = {New Requirements – Existing Solutions}
Cost Savings = Existing Solutions

Currently, approximately 80% of the workstations are redesigned against new / changed product, while actual requirement of redesign is approximately 20-25%. Therefore, the introduction of the ALDIMS tool envisages huge potential for cost savings.

Current practices results savings / line = 20-25%
Potential achievable savings = 75-80%
Enhancement in savings = (75 – 25) = 50% minimum savings

As the 'Δ' may change from program to program, therefore, a reasonably acceptable value is taken for a range of product variants and new programs over one year period. The current savings can be enhanced by at least 50% through knowledge based approach to line design / reconfiguration process. Design cost analysis of the 'As-Is' and 'To-Be' approaches is provided in Table 27.

<table>
<tr><th colspan="6">Cost Facet</th></tr>
<tr><td colspan="6">For Engine Variants – Major Impact on Design and Build of Machines</td></tr>
<tr><td><b>1</b></td><td colspan="4"><b>As-Is</b></td><td><b>Approximate Values</b></td></tr>
<tr><td></td><td colspan="4">Total line cost</td><td>£30M</td></tr>
<tr><td></td><td colspan="4">Actual machine build cost</td><td>£18 M</td></tr>
<tr><td></td><td>Machine</td><td>automatic</td><td>semi-automatic</td><td>manual</td><td>£0.62 M</td></tr>
</table>

	design cost	£4.1k	£4.9k	£3.4k	£4.13k x 150 = 620k
		Average = (4.1 + 4.9 + 3.4) ÷ 3 = £4.13k			
	Installation and commissioning cost {25% of total cost (VDW, 1999; average value at Ford)}				£7.5 M
	Controls cost 10% of total cost (from Krause GmbH)				£3 M
	Virtual verification Cost {changes + installation}				0
2	**To-Be**				
	Virtual Engineering Cost added (CCE + ALDIMS) For CCE = 20hrs@£200/hr = 4k				*£0.60 M* *4k* x *150*
	This extra cost helped to evaluate product-resource constraints rapidly and make quick decisions about machine configurations				
	For a changed product, Δ = 20-25% hence the prospects of reuse 75-80% but currently, only 20-25% are reused, hence atleast 50% can be reused				
	Cost savings in machine build 50% of 18M				*£9 M*
	Cost savings in machine design 50% of 0.62				£0.31 M
	Installation and commissioning cost*				£7.5 M
	Controls cost				£3 M
	Total Line Cost {0.6+9+0.31+7.5+3}				£20.41 M
	Potential Cost Savings (machine design + build)				***£9.59M***
	Total percentage cost savings				***68%***

*Virtual commissioning cost also has a significant impact and the analysis is provided in the next section

Table 27: Summary of the machine design and build cost savings 'As-Is' and 'To-Be' approaches

Cost savings in installation and commissioning

Currently, all production machines at Ford are commissioned physically. Labour-intensive commissioning and late machine changes can account for 25% of the machine costs[81]. An investigation[82] for the German Association of Machine Tool Builders (VDW[83]) showed that the commissioning phase of a production system accounts for up to a quarter of the total project cycle time, and in turn, cost (VDW, 1999). The BDA project has made a case for complete virtual verification of the entire assembly line. However, there are some issues which need to be manually verified on physical stations. On average, there are 20% of the commissioning tasks which need to be manually verified. Typically workstations cost an average of £400k, therefore, £100k is the average cost of installation and commissioning. With the help of CCE and ALDIMS

[81] Ford's Fox powertrain assembly line virtual engineering cost estimation exercise

[82] Report on cross-project planning of complex machinery and equipment

[83] German Machine Tool Builders' Association – Verein Deutscher Werkzeugmaschinenhersteller (VDW)

tools, Virtual Commissioning (VC) can be carried out for the entire assembly line. The concept of VC is to engineer a virtual system, often based on 3D dynamic representation, in an attempt to foresee design errors and inconsistencies before large resources are invested in implementing a physical system. The VC ensures system's real time dynamic behaviour in virtual environment. Once virtual commissioning is complete then the simulation results may be transformed into rules in ALDIMS. The commissioning cost savings are depicted in Table 28.

Cost Facet			
For new engine programs – Major impact is on commissioning phase			
1	**As-Is**		
	No Virtual verification; Physical verification of product-resource constraints; Costly changes; Installation and commissioning problems		
	Physical build cost	£400k/machine	
	Design cost (virtual build, 3D model, all CAD put together and minor revisions)	£4k	
	Commissioning cost is approximately 25% of total cost This cost may be considerably reduced in the To-Be approach	£100k	
2	**To-Be**		
i	**CCE only** – Complete virtual verification of assembly line machines		
	Virtual Commissioning	**Virtual Verification**	**Physical Verification**
	Possible percentage virtual commissioning verification	80%	20%
	Approximate Cost Value	£80k	£20k
	Virtual commissioning will cost CCE takes 20hr to virtually verify; 20@£200/hr = £4000	£4k	£20k
	Total cost with CCE tool	£4k + £20k = £24k	
ii	**ALDIMS + CCE** (parameterised analysis and reduced virtual verifications)		
	Typically there are 80% resources which need no or slight modifications therefore virtual verification not required	80% of £4k	£20k
	Additional Cost for parameterised analysis ALDIMS takes 5-10 min for parameterised analysis; 0.25@£200/hr = £50	Negligible	
	Reduced virtual verification	£0.8k {4k – (80%of4k)}	£20k
	Total cost / machine with the combined use of CCE and ALDIMS tools	(0.8k + 20k) = 20.8k	

	Total cost savings / machine	100k – 20.8k = **£79.2k**
	Approximate cost saving for entire line with combined use of CCE and ALDIMS tool; (Fully Automatic and semi-automatic machines = 80 approximately)	
	Total cost saving / line	£79.2k x 80 = 6336k = £6.3M
	Current Commissioning cost	£100k x 80 machines = £8M
	Total %age cost savings Combined use of CCE and ALDIMS tools during the commissioning phase	**78.75%** (6.3÷8)x100

Table 28: Commissioning Cost savings – 'As-Is' vs 'To-Be'

Therefore, as much as 78% of the commissioning cost can be reduced by a combined use of CCE and ALDIMS tools. The machine suppliers normally quote for the entire assembly line and for complete workstations on the assembly line. Typically there are many workstations / modules and components of workstations (75-80%), which can be used without any alteration or with slight modifications in control programs and/or modifications in existing configuration of the workstations/modules. As this information is not readily available, therefore, the automotive OEMs have no control over the overall price of the new / changed assembly line. The price of the assembly line is controlled and dictated by the supply vendors. A combined use of ALDIMS and the CCE tool will ensure a rapid analysis of changes required on each workstation and, in turn, on the entire assembly line, therefore, the automotive OEMs can ask for the quotation for the changes required on the existing design of the line rather than complete redesign of assembly line. Thus, there is a potential for enormous monetary savings and shift of price control from machine vendors to automotive OEMs.

On the other hand, the costs associated with manual verifications on physical resources could result in expensive amendments in BoP and/or BoR and even product specifications. The additional time spent at earlier phases of program management and developing rule-based knowledge support is well justified when compared to the repeated time losses and cost augments against frequent product and consequent machine changes.

The existing approach results in inconsistent costs, time delayed achievements and lack of confidence until the ramp-up phase of the line, therefore the powertrain assembly line build and commission time varies a lot, the average time is five to six months. In some cases, the constraints appraised manually on the physical workstations can be extremely costly as any assembly hard point at this stage implies rebuilding the machine or reconsidering BoP or even redesigning product specifications. The new approach provides a confident 'design and build', consistent time and costs, confident installation and error free commissioning and ramp-up phase with increased confidence, thus, avoiding any unnecessary and unexpected surprises at the later stages of new / changed programs which may incur huge losses, a maximum time of line build and design is calculated to be less than four months. Therefore, there is a definite potential of time and cost savings within automotive OEMs by adopting knowledge based design approach through ALDIMS tool.

7.3.3. Quality Facet

The ALDIMS tool indirectly affects the quality of the engineering process. The machine design / retrofitting activity on a powertrain assembly line is a perpetual process, therefore, the virtual line designing and product-resource constraints evaluation is also continual. This highlights the importance of a generic line cloning and automatic evaluation of the product-resource constraints. There are some latent improvements and benefits of the new approach directly affecting organisational capabilities in terms of assembly line design / reconfiguration and provide competitive advantage in the long run. A quick engine fit analysis will ensure rapid product adjustments and cost effective decisions especially at earlier stages of the program management. One of the real issues faced by almost all the automotive OEMs is heavy dependence on the supply chain partners particularly for the machine design and build. The current virtual engineering tools and the knowledge based line design move the control of line design activity from machine builder to automotive OEMs and ensure the value adding activities are in the control of automotive OEMs rather than the suppliers. This shift in control of price can have extensive impacts not only in significant reduction in assembly line price but also in budget reallocation and rationalisation to spend money in more innovative and value adding activities.

After the introduction of the ALDIMS tool, lower level of expertise for the said activity is required as greater use of available knowledge is ensured. The novice engineers can confidently proceed consulting the system while the domain experts can concentrate on more critical tasks. There are approximately six to eight domain experts working on the virtual verification process duly supported by requisite staff. The CCE and the ALDIMS tools will ensure rapid virtual verification and constraint evaluation. As 75-80% of the resources will be reused in their current configuration state or with slight modifications, the domain experts can concentrate on the remaining 20-25% of the resources which need to redesigned/reconfigured. Critical design decisions which may consume a lot of time can be processed automatically thus reducing manual efforts, therefore, the domain experts can spend more time on value adding tasks. The ALDIMS tool also helps guiding the domain experts as to what needs their immediate attention e.g. the resources categorised as redesign should be designed and evaluated first, then the resources to be reconfigured, thus the tasks are prioritised.

The quality related hidden benefits are summarised in Table 29.

Quality Facet	
As-Is	**To-Be**
Judgemental selection of potentially problematic stations, inefficient utilisation of expertise	Domain experts can efficiently utilise their time by focussing on new machine design / complex decision making
The workstation is treated a complete unit and equal time is spent on each module / component	Modular approach ensures further focus on specific modules / components out of many on a single workstation
New line design for new programs Repetitive constraint evaluation for engine variants	Cloning a line from generic line will ensure standardisation of BoP and, in turn, BoR and successive improvement in quality of the design and build of workstations
Lack of explicit representation methods for decision recording	Complex decisions management and recording will ensure avoiding repetition of mistakes again and again and reuse previous knowledge of programs

Table 29: Summary of quality facet 'As-Is' vs 'To-Be' Approaches

7.4 Evaluation of Prototype Design Tool

The developed prototype system is evaluated in terms of execution of performance (service time and capacity of information handling), development and set up effort, rule formulation, skill level required and accuracy of results of ALDIMS compared with the domain expert.

7.4.1. Performance

Performance refers to responsiveness or throughput of the system[84] i.e. execution time of a component of a software application in order to provide the requisite service to the system users. The ALDIMS is tested for performance against user requirements in terms of time and capacity of the system to process and handle large amount of data (e.g. number of rules on a certain workstation). Unlike other ontology languages, as many as 55 arbitrary rules on one of the

[84] The throughput of a system is the number of events processed in a specified interval of time

workstation (OP1860) were tested and executed successfully by the inference engine of the WSMT tool. Thus the ALDIMS is stable in terms of capacity to handle large quantity of rules and concepts. As the new approach is based on parameterised constraint evaluation, the results can be obtained in significantly reduced time. Detailed time study for ALDIMS is still to be carried out, however, initial usage of the tool shows that activities such as adding concepts, formulating axioms and data mediation can each be carried out in less than 10 minutes e.g. during the evaluation of typical assembly machine.

7.4.2. Design, Development and Set-up Effort

The ALDIMS transforms the results of labour-intensive and time-consuming simulation activity into reusable, rapid, constraint-based analysis. However, the effort of making rules and axioms, for the first time, is more than that of simulation based analysis because the rules / axioms formulation would require complete simulation, assessing KCs and then formulating rules and axioms. On the average, there can be 20-25 rules on a particular station which may take an additional of four to five hours maximum, once the knowledge engineer and the domain expert work in collaboration. As the total number of workstations on a powertrain assembly line is 150, therefore, the knowledge based system in its present form will have approximately 3500 rules in total, consuming an extra time of 750 – 800 hours. Nevertheless, the rules and axioms will be made only once and the same rules will be applicable for the lifetime of the machines / line and even for subsequent machines and lines. The effort of making rules is one time and the benefit of the effort is spread for succeeding machines, programs and lines. The knowledge based approach promises the reuse of the available knowledge, therefore, for any next product change, the engineers do not have to carry out all the 3D model simulations again and again.

7.4.3. Rule Formulation

The rule making requires careful analysis of the simulation activity and defining KCs in terms of product-resource constraints dependency parameters. It is advisable that knowledge engineer and simulation engineer / domain experts work together to transform simulation results into reusable axioms based on parameters of products and resources. Defining KCs and formulating rules based on decision logic of domain experts is not simple, however, the transformation of the results of simulation analyses into rules is uncomplicated. The Logic Programming (LP) syntax used in the WSML is easy to understand and implement especially defining and structuring rules / axioms is easy. The KCs are translated into local variables (private) within the axioms. The conditions are applied by simple 'If-Then' statements within the axiom, as described in chapter 4, section 4.3.3.2.

7.4.4. Skill Level Required

Skill level for using the system is basic acquaintance with the windows environment and skill level required for designing / editing the developed system understanding of the basics of KB system, LP syntax and WSML ontology language. WSML ontology language has normative syntax and easiest to learn and understand while LP syntax can also be learnt with some training. The WSMT editor helps editing the syntax in both WSML language and LP syntax, therefore, the process becomes easier in the WSMT editor.

7.4.5. Accuracy of the ALDIMS' Results

The results of the ALDIMS depend upon the level of detail of the KCs and the accuracy of the rules formulated. In order to check the effectiveness of the formulated rules, a comparative analysis session was arranged in which OP 60 and OP1900 workstations were investigated in order to verify the outcomes of ALDIMS tool with that of observations of the virtual engineering application engineer. The outcome of the ALDIMS tool for the said two workstations remained consistent with the simulation results obtained by the domain expert (the formulated rules are mainly derived from the simulation results), therefore, the KB system is reliable enough to be consulted for critical decision making. However, if the logic of the rules is flawed or the KCs are not properly selected then the outcome will be wrong. A critical comparison of the observations of the virtual engineering application engineer against results from the ALDIMS tool are provided in Table 30.

S. no.	KCs on Workstation	Observations of the VE Application Engineer	ALDIMS Output
1	**OP60**		
	Engine slide plate	Amendments required	Reconfigure
	Engine mounting plate	No changes required	Use-as-is
	Nut runners	No changes required	Use-as-is
	Dowel Pins	No changes required	Use-as-is
	Fixture clashes	Workstation guarding gets too close to the product while engine slide plate rotates	None
	Tool access	Okay	Okay
	Operator access	Okay	Okay
	Result: Assembly hard points	**01**	**01**
2	**OP1900**		
	OP1900 Nut Runners	Retrofitting and then Reprogramming is required	Reconfigure and Reprogram
	Fixture clashes	None	None

	Tool access	Okay	Okay
	OP1900 System	Use-as-is	Use-as-is
	Result: Assembly hard points	**01**	**01**

Table 30: Comparison of results - Domain expert vs ALDIMS tool

The decision support provided by the ALDIMS tool is consistent with the observations of the virtual engineering application engineer. The fixture clashes, tool access and operator access are defined in terms of automation system's overall assembly hard points which in turn depend upon product length, width and height. It has been observed that the execution of the defined rules cannot be wrong, the most important aspect is the selection of right product-resource KCs on a particular workstation for a certain product build stage. As described in chapter 6, section 6.5 and section 6.7.4, the knowledge reuse infrastructure for assembly line design is useful and viable, however, the refinement of rules is also important along with selection of right KCs. The total number of KCs for a certain product build stage provides confidence and refinement to the results of the ALDIMS tool.

7.5 Strengths and Weaknesses of ALDIMS

7.5.1. Strengths

The ALDIMS application provides rapid parameterised analysis of product change effects on automation resources and aims to reduce virtual verification activity. The ALDIMS tool can help in effective distribution of expertise across engineering domain by pinpointing the problematic issues to be dealt by domain experts while less critical issues may be solved by the novice engineers. The system provides an open (no exclusive rights to the software, availability of the source code and the right to be able to modify it[85]), scalable (immunity to changes in the form of addition, deletion or modification) and extendable (provision for the addition of new functionality, enhancement through increased capacity and expansion mechanisms) architecture for knowledge preservation, improvement and management. The system is web accessible and multiple users can simultaneously access the system.

[85] http://oss-broschuere.berlios.de/broschuere/broschuere-en.html#N3142

7.5.2. *Weaknesses*

The ALDIMS tool, though supported by inference mechanisms, lacks common sense; it cannot make creative decisions in unusual circumstances rather it manipulates whatever information is available to the system. A slight human error in formulation of rules may lead to inaccurate results and thus erroneous decisions. The system is brittle in response to adapt to unpredictable industrial situations, thus, may need to be updated against changing business requirements. Another limitation of the ALDIMS tool is that it views the world as consisting of truths and falsities only, which is not always correct. There are many scenarios which are fuzzy enough to be dealt with evolutionary computing techniques, which have not been taken into account in the current prototype. Similarly, the reliability of the automaton resources is considered optimum when selecting / designing a new resource. The knowledge collection and interpretation into rules is not easy and needs team work of the domain experts and knowledge engineer. Another weakness of the system is that only those characteristics are dealt which can be parameterised.

The ALDIMS tool is used to compare the product-resource constraints which can be parameterised. There are a few features of product / resource which may not be readily converted to numeric values e.g. profile of the product at OP1860 station (unless several points are compared in relation to spatial reference). This is one of the inherent weaknesses of the parameterised approach. The rules are formed on the basis of a rationally plausible product envelope. The rules will not work for any radical change in the engine and/or assembly machines for the engine. For example, if the engine changes from an internal combustion to an electric engine, the knowledge based system, in its current form, will fail to detect true effects of the functional and design changes of the product on existing resources. The KB system currently is applicable for only the dimensional / geometrical changes of the engine and not for the internal design or functional changes of the engine. Similarly, the resource configuration is assumed to be consistent to the conventional norms / standard reference point, e.g. the nut runners' configuration on a specific workstation is almost always same. The system, currently, cannot determine the spatial reference of the resource and its smaller units. For detailed analysis, 3D model-based investigation will have to be carried out especially for the workstations / smaller units which need change in configuration / redesigning.

The benefits and limitations of the ALDIMS tool are summarised in Table 31.

	Benefits of ALDIMS tool	**Limitations of ALDIMS tool**
1	Rapid evaluation of product-resource constraints	Rules require iterations to develop confidence, formulating rules can be time consuming

2	Easy to learn and use	Hides crucial concepts and presents a black-box approach
3	Parameterised analysis	Can never replace 3D model based simulations
4	Ensure reuse and management of knowledge	Knowledge elicitation can be difficult e.g. selecting and defining KCs on a particular station is not straight forward
5	Results of simulations are recorded in the form of reusable rules	Not every KC and feature can be parameterised and translated into rules
6	Explicit mapping and relations among major concepts i.e. PPR	Presents a black-box analysis hiding crucial details for detailed analysis
7	Potential to extend to other domains especially cost issues	Lacks common sense for odd situations

Table 31: Summary of benefits and limitations of ALDIMS tool

7.6 Limitations of the Prototype Tool

The ALDIMS tool is designed and developed for decision support in the line design / reconfiguration only, and the directly related and mutually affecting domain entities i.e. PPR, have been considered. There are many indirectly related entities and activities which though have not been considered, however, can be included in the flexible architecture. These activities include productivity analysis, overall program management, ergonomics analysis, plant layout, quality issues, machine cycle times, related business processes etc. though it does evaluate the research benefits on some of these business processes.

The user interface of the prototype system is rather basic with limited functionalities. As the purpose of the prototype system is to prove the applicability of the new approach and is meant for test implementation therefore more time was spent in developing core functionalities and focusing on technicalities of the system rather than presentation of the tool. It is however intended to improve the front end in near future and is one of the tasks suggested for the future work.

One important aspect i.e. the process logic (of control programs[86]) for the automation resources is not included. Control programs are one of the important parameters in the design and

[86] Control programs of automation resources consist of component logic and process logic

reconfiguration of the automation resources, however, due to unavailability of first hand information, it has been left as a future task. The system has a narrow focus of assembly line design and reconfiguration activity in a wide engineering perspective in an automotive industry. However, keeping in view the complexity of the assembly line design / reconfiguration activity, the narrow focus of the developed system is advantageous as well. Keeping the system focussed for a particular activity will provide acquaintance and uptake time for the stake holders.

Though the KB system is limited to line design / reconfiguration, however, the developed system can be easily extended, unlike dedicated expert systems which require huge effort, it is easy to scale up the ontological knowledge based system. Lack of knowledge for control programs of the automation resources is realised to be a limitation, however, the current infrastructure is capable to include it in near future.

7.7 Summary

The chapter has described evaluation methods of the developed system in order to testify the accuracy of the system and assess its efficiency and effectiveness in terms of time, cost and quality. The evaluation starts with a review in consideration of the impact of the existing and new approaches in the context of the Ford's Global Product Development System (GPDS) with existing and new approach, highlighting key differences and potential impacts of the new approach on the GPDS. This was followed by time, cost and quality comparisons, evaluation of ALDIMS prototype tool, implementation effort and viability of the approach. A review of changes in practices / approach was also summarised and finally the limitations of the prototype tool were described.

CHAPTER 8. CONCLUSIONS

8.1 Conclusions

Driven by the need, this research has contributed to the provision of knowledge based support for the virtual engineering tools in the assembly line design and verification process. The modular virtual engineering application (CCE tool) provides verified, reusable machine mechanisms, however, to more efficiently and effectively utilise these modular resources, a knowledge based infrastructure support is necessary which is realised in the author's research. The knowledge based support will help verify relational constraints rapidly and provide decision support as to what should be verified virtually. In addition to this, the knowledge based support is provided in case new automation resource is to be designed by efficiently selecting available resources or effectively modifying the existing resource. Best utilisation of the expertise of domain experts is ensured by semi-automating some of the manual effort and ensuring reduced simulation activity. Similarly, optimum time allocation of the team members for a focused constraint analysis is assured.

The research concept is proved through design and development of an ontological knowledge based system and its practicality in real industrial situations is measured through test implementation at Ford. The case studies prove that it is viable to adopt this new approach for assembly line design / reconfiguration.

The principal objective of the research was to design, develop and prototype a cross-functional knowledge based system in order to provide a more efficient modular approach for the rapid design / reconfiguration of assembly lines. This was to be achieved through parameterised analysis among the directly affecting PPR entities. The objective has been achieved in the form of a KB system called i.e. the *ALDIMS* tool. The initial research hypothesis set out in the beginning of the research was that the responsiveness capability of automation resources against perpetual product changes can be enhanced by utilising knowledge based system which has successfully been demonstrated, therefore, it is strongly advocated that the new approach can be adopted as well as enhanced by further research activities.

The research directly addressed the automotive industry requirement of rapid *engine fit analysis* which has been made parameterised. To help achieve this time compressed analysis, a few necessary support activities have been included in the system. These support activities helped realise engine fit analysis with reduced manual interactions and effective utilisation of available knowledge. A brief of the research goals and research outcome is summarised in the Table 32.

	Research Goals	**Research Results**
1	Rapid constraint evaluation for engine fit analysis	Parameterised analysis of constraints
2	Reduction in simulation based analysis	Categories of change, what is required to verify through simulation based analysis
3	Smooth interaction and common understanding	Data mediation
4	Machine design support	Library characteristics, efficient selection of requisite resource
5	Assembly line visualisation	Hierarchical classification
	New knowledge from available information	Ontological inferences and axioms

Table 32: Research goals and results

The work carried out during this research has been recognised internationally and most of the research results have been published in refereed scientific journals and peer reviewed conference proceedings, a brief of which is described in Chapter 1, section 1.6. This section provides a brief of the scientific contributions. There are four major scientific contributions made and described in journals and conference proceedings.

1. The concept of transforming legacy software applications to knowledge based systems by introducing a knowledge layer based upon ontology wrapped around the existing software applications is revealed and proved[87,88].
2. An initial ingest architecture for a knowledge based assembly line design, assembly automation resource structure and knowledge modeling is designed, developed and exploited[89]. A combined use of inferential and deductive logics is described through ontological axioms and the concept refinement rules in the knowledge based system.
3. The application of ontologies for embedded web services at the factory floor in terms of integration within a distributed computing infrastructure is shown at elementary level, the

[87] http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5356779
[88] http://www.dirf.org/jism/2.pdf
[89] http://ceur-ws.org/Vol-748/paper3.pdf

concept was proved on the Festo Rig with semantic web services to control the imitating assembly operations[90].

4. The real industrial challenge of rapid constraint evaluation for designing and/or reconfiguration of powertrain assembly lines to cater for a new/changed product, with the current state of the line design process and enhancements through knowledge based system, is presented [91].

8.2 Contributions to Knowledge

The principal contribution here is the creation of a rapid constraint evaluation method to complement the CCE virtual engineering toolset, which is the next logical step in further improving the efficiency of the process once the entire assembly line is virtually designed and verified. A 'proof of concept' knowledge based system has been designed, developed, prototyped and test-implemented in the operational context of the impact of product change on existing machine tools on an engine assembly line with classification of category of changes held in the knowledge based system. The work conducted during this research has been mainly focused on parameterised analysis of the product-resource constraints in order to quickly evaluate the changes required in the automation resources against new / changed engine variants. An innovative knowledge reuse infrastructure for rapid parameterised constraint evaluation of reconfigurable automation resources on an engine assembly line has been developed and evaluated for potential future full-scale implementation at automotive OEMs. This contains the capability of classifying categories of the changes required in the automation resources against new / changed products. The new approach has been realised in the form of the ALDIMS tool.

8.2.1. New Methodology

Rapid Parameterised Constraint Evaluation

Presently, the product-resource constraints are checked manually through physical studies or simulation via 3D models. The author has devised parameterised analysis of the constraints by defining KCs of products and resources and formulating rules and axioms on the defined KCs thus eliminating or reducing the physical or virtual verification activities for many of the workstations on the assembly line. Automatic evaluation of PPR constraints along with modular

[90] https://www.inderscience.com/IJSCOM
[91] http://airccse.org/journal/ijdkp/papers/0911ijdkp03.pdf

approach in assembly line perspective is a novel approach, there is no in-house option available as well as there is no known commercially available solution for it.

Product-Resource Key Characteristics

The success of the developed system depends upon the selection of the proper KCs of correct product stage and the corresponding assembly machine. The idea of checking the assemblability of product parts on respective resources by formulating rules on KCs and checking the results in terms of results of defined rules is believed to be a novel approach in assembly line design and reconfiguration.

Categorisation of Change

The developed system provides specific recommendations for specific components. There are four possibilities for alterations in the machine components in response to the product changes, namely: *use-as-is, reprogram, reconfigure,* and *redesign*. The system evaluates the constraint conditions and recommends the changes required in the resources, as assessed. There is currently no known KB system providing specific recommendations for specific components of machines in the engine assembly domain.

KB System Complementing CB Technology

It is believed that currently there is no known KB system developed for the support of simulation tools such as CCE tool that supports CB technology applied to the design of assembly automation resources. The ontology of an engine assembly line has been studied previously (Lastra and Delamer, 2006; Alizon et al, 2006; Xuemei, 2007; Berger et al, 2008; Chakrabarty et al, 2009) however, the ontology of the PPR domains along with CB technology including inter-dependency rules and axioms is thought to be unique to this thesis and is considered the first ever effort to amalgamate knowledge engineering, ontology engineering and component based technology in the assembly automation resource design and reconfiguration activity.

8.2.2. New Technology

The ALDIMS Tool

The research concept is successfully demonstrated through an ontological knowledge based application to help achieve a more rapid and robust approach towards assembly line planning. The ALDIMS tool contains cross-functional dependency relations, explicitly formulated in the form of rules and axioms, among PPR domains especially creating relations for smaller units of machines and atomic process steps in terms of assemblability of product parts. As it is believed that there is no modeling and simulation design tool mature enough to be fully used in real industrial challenges, therefore, this is the first KB system which complements a recently operational virtual engineering tool based upon CB technology in the automotive industry. An open, platform independent, scalable and interoperable knowledge application layer has been developed which helps achieve interoperability by seamless flow of information among domain software applications. Designing and developing a knowledge based system on ontological foundations provides taxonomical and reasoning capabilities in addition to conventional user defined rules and axioms.

Currently the outcome of modeling and simulation activity has no infrastructure to record the results other than simple descriptive documents. Once the tedious and time consuming simulation is carried out with CCE virtual engineering toolset to evaluate the product-machine constraints and detect assembly hard points, the results are left largely unrecorded and not readily reusable. The decisions are made on the basis of simulation outcome, however, the same activity would have to be carried out in case of product change. Simulation activity is carried out to make decisions, the decisions are made entirely based on the human judgement and experience. The simulation process does not make decisions, it helps the engineers to come to a conclusion and make decisions. The knowledge generated during simulation is not captured effectively hence the vital knowledge remains latent or is lost. The ALDIMS knowledge based system will help to capture the knowledge generated during the simulation activity in the form of rules and axioms, record decisions and help future decision support activity. Hence the ALDIMS tool has introduced decisions management and decision support for the line design / reconfiguration activity.

8.2.3. New Engineering Workflow

The thesis has shown in proof-of-concept form, through a set of example workstations, that the new approach is potentially useful in nontrivial cases and thus will result in new engineering workflow. The new workflow, discussed in chapter 7, section 7.2.4, will shift the control of the cost of machines from vendors to the end users (automotive OEMs) as well as dependence of automotive OEMs on the machine suppliers will be reduced. Currently, the design ownership is held by the machine builders, order is placed for the complete assembly line and machine vendors dictate the overall price and the delivery time of the machines. With the introduction of the knowledge based support, potential change in relationship and interactions between the automotive OEMs and the machine suppliers can be realised especially the control of machine design activity in the hands of automotive OEMs and order placement for only the changes required (‘Δ’) in the existing machines on the assembly line rather than the entire assembly line.

8.3 Industrial Contributions

There is currently no commercially available tool specifically designed for the rapid decision support during assembly line design / reconfiguration activity. This ‘proof of concept’ research has demonstrated that by creating an adaptive knowledge model for PPR interdependency constraints, an effective decision support system can be formulated with multiple advantages including:

- Enabling agility, scalability and transparency across engineering domain
- Automated response to product change
- Decision support emulating human consultant
- Industrial perception about uses and benefits of ontology
- Raising the awareness and understanding of the complex activity of assembly line design / reconfiguration

The integrated knowledge modeling of the independent domains, though not novel in itself, is extended to provide a rigorous foundation to build knowledge management framework and support a shift from manual, judgemental-based assumptions to concrete, confident and evidence-based decisions for assembly line design and reconfiguration activity.

The system potentially provides universal access to collaborating companies, therefore, third party vendors can readily access the knowledge which is structured in the KB system thus allowing enhanced facilities for knowledge capture, exploitation and reuse. The ALDIMS tool offers an infrastructure in linking the wider knowledge management strategy with a KB system for a CB virtual engineering tool for automotive sector. The automotive OEMs and their vendors will directly benefit an integrated PPR infrastructure, business readiness through predictability and promptness and improved communication.

8.4 Future Work

The future work consists of reducing weaknesses and limitations as well as enhancing functionalities and strengths.

8.4.1. Robustness of the Approach

The newly developed parameterised approach is tested initially with Festo rig in an experimental set up and later for two different engines passing down a common assembly line. It is believed that finalisation of rules should be made with several combination of sizes within plausible product envelope so that a refined version of rules may be obtained. The ALDIMS tool has not been tested in terms of robustness of the approach. It is suggested to now carry out a robustness analysis and this is recommended as one of the future tasks in order to acquire greater confidence in the new approach. The risks associated in adopting the parameterised approach can be minimised by putting greater efforts in defining KCs of product-resource constraints. Given that, the correct execution of the defined rules is robust enough, therefore, the success of the approach depends upon defining the correct KCs on each workstation for the respective product build stage and then translating the KCs into rules and axioms. The rules may be refined with historical data of changes in engine sizes and shapes and further predictions may also be derived on potential future changes. The effectiveness of the rules can be improved by iterative analysis of the product change effects and, thus, improve the robustness factor. It is recommended to carry out iterative analysis with lots of data for the changed engine systems to get refinement in the rule formulation and get confidence in the approach.

8.4.2. Next Lifecycle Stage of ALDIMS

The main purpose of the ALDIMS prototype system was to test-implement a knowledge based solution for the decision support during 'engine fit analysis' in the context of assembly line design and reconfiguration. The ALDIMS tool has proved the initial concept of rapid analysis of constraints along with a range of allied functionalities. The concept is ready for initial implementation at Ford first, and then, a full scale implementation on the whole assembly line.

BDA Project Engineering Tools			
1. CCE Tool	**Current Status**	**2. ALDIMS**	**Current Status**
Concept Ready	Achieved	Concept Ready	Achieved
Implementation Ready	Achieved	Implementation Ready	Yet to Achieve
Implementation on whole line	Yet to Achieve	Implementation on whole line	Yet to Achieve

Table 33: Lifecycle Stage of the CCE tool and the ALDIMS

The Table 33 provides brief of the lifecycle stages of the new engineering tools developed during the BDA project. The next step is to work on the 'implementation ready' phase of the developed knowledge based system. Based on the current phase of the developed system i.e. 'proof of concept', the ALDIMS prototype is the starting point for the introduction of knowledge engineering principles in complex decisions during assembly system design. The following steps are recommended to be followed as a future work:

i. Design and engineer a new version of the system with the end users
ii. Populate the rule-set with domain experts for the entire assembly line
iii. Evaluation in real engineering process with Ford engineers

After the 'implementation ready' phase is completed, the system may be extended to be used on the entire assembly line as well as on different engine production plants at automotive OEMs.

8.4.3. *Automatic Population of ALDIMS*

Currently, the ALDIMS tool has to be manually populated with information. Future research envisions populating the KB system automatically. The main idea is to bring the related concepts from legacy applications into the ontology in the structured form and then apply rules and axioms on the imported instances keeping both the phenomenon automatic i.e. data extraction and rule application and the provision for the same is achievable in the developed system as shown in the Figure 88.

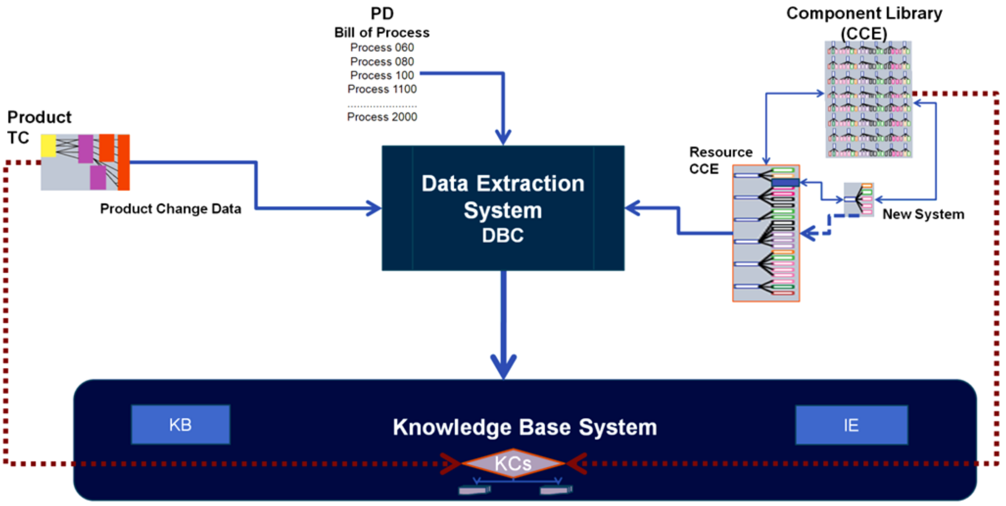


Figure 88: Ideal scenario for KB interactions and ontology populating

For the process of populating information in the KB system to be made automatic, ontology and legacy applications need to communicate. This is possible through Database Connectivity (DBC) option in the form of Application Programming Interface (API) provided in the commercial software applications. With the help of DBC, it is possible to establish knowledge mapping between the meta-data concepts of the participating systems thus making the data extraction process automatic.

8.4.4. *Automatic Ontology Update*

The BDA project results have already been utilised in the form of the CCE virtual tool in the automotive sector e.g. at Ford Company. It is envisaged that the CCE toolset will soon be integrated in the commercial engineering workflow at Ford. Therefore, the CCE tool will progressively add automation resources, modules and components to form a reference library of automation resources. On the BDA project, this has been termed as 'Gold Standard Library' - GSL. As the CCE tool increases its database of resources, the ontology needs to be updated alongside it. One of the advantages of the CCE tool is that it can provide output of the 3D modelled resources in XML format. Therefore, the XML output of the CCE tool and the existing ontology can be readily compared and the ontology of the automation resources can be updated automatically as shown in Figure 89.

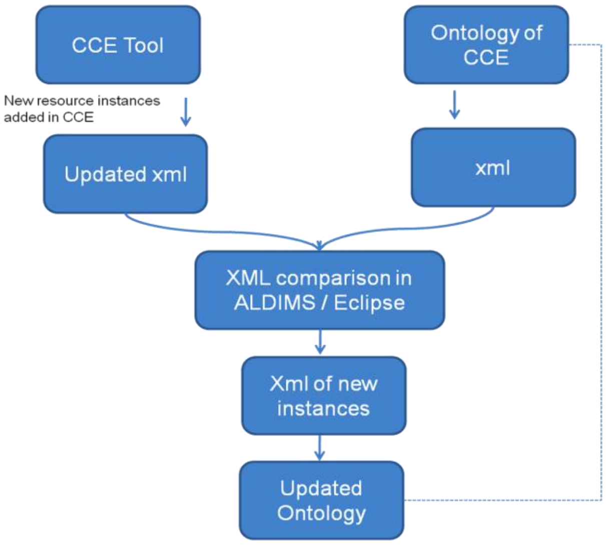


Figure 89: Automatic ontology update strategy

This concept was tested during the implementation of the system for the Festo rig where the ontology was updated by matching CCE's XML with that of Festo Rig's XML, the following steps were to update the ontology were defined.

- Read 'CCE XML' and extract required information
- Read 'Festo XML through OWL Ontology' and extract required information
- Compare the two files, detect differences and update ontology

This test was implemented in protégé ontology editor. Future work would integrate the applications and automate the update process.

8.4.5. From Configuration to Runtime Environment

The Figure 90 illustrates how the implementation of Semantic Web Services (SWSs) can be carried out at Ford with each SWS linking into a variety of existing services thus helping to create a Service-oriented Architecture (SoA). A Service-oriented Architecture is a set of architectural tenets for building autonomous yet interoperable systems (James and Smit, 2005). The knowledge present in the ontologies has led to a distinct focus on configuration and line runtime management.

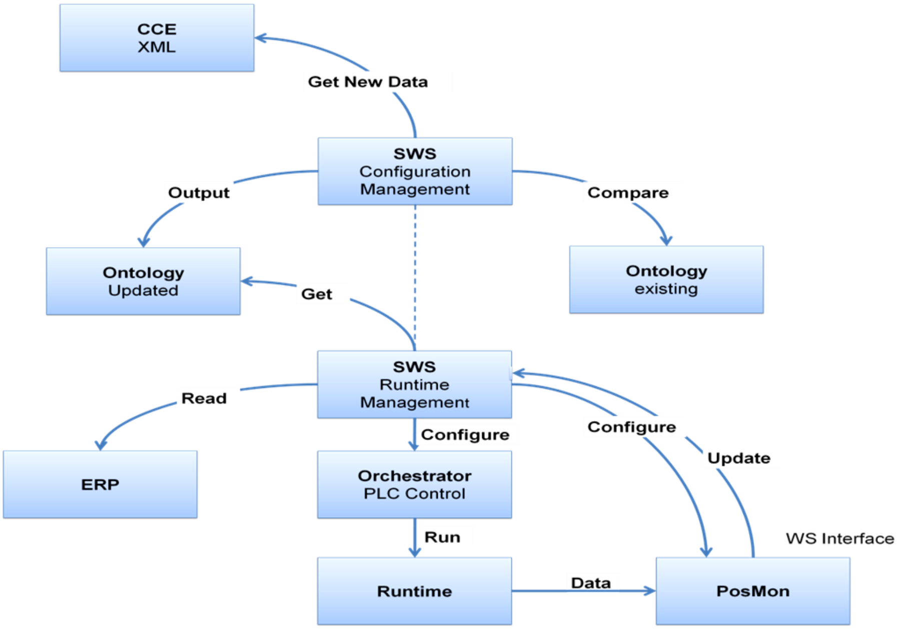


Figure 90: Application of Semantic Web Services (SWS)

The Figure 90 shows the possibility of using ontology in the runtime environment. To use ontology for the runtime management, web services can be defined as for configuration management and for runtime management. Web services for the configuration management would get updated information of the line from CCE tool / Process Planner and update the ontology of the line. Once updated ontology is obtained, it can be linked to POSMon[92], ERP[93] or CRM system etc. through SWSs to achieve business objectives.

For example if there is a jam on a conveyor, this information can be used by the ontology of the line to aid in the diagnosis of errors in the system. If there are errors, the SWSs can use the line ontology to find a remedy to treat the error. This process automates some of the response provided by a production engineer. Using the ontology the KB system (through PLM service) can instruct the control mechanism to notify dependant stations on the line that an error has occurred and even request a halt in the production.

In addition to this, the use of the ontology will enable the notification of other appropriate services in the supply chain e.g. CRM and ERP. For example, the ontology may be used to order a replacement part for the line by interrogating the components affected by the error. A supporting knowledge base of previous faults linked to probable cause could aid in this process. The PLM could also notify other services such as CRM and ERP of issues on the line and therefore potential impacts on production output.

8.4.6. Improved Front End of the KB System

As described before, the user interface of the prototype system is rather basic. The front end of any software application should be user friendly and easy to navigate. It is recommended to improve the user interface to make it more interactive as the system is enhanced and improved during the next ‘implementation ready’ stage of the ALDIMS tool development.

8.4.7. Cost Estimation Functions

Previous research results have indicated that over 70 per cent of the production costs of a product are typically determined during the conceptual design stage (Asiedu and Gu, 1998; Shehab and Abdalla, 2002). As the production systems design is agreed, the cost estimation is also typically with template sheets containing relationships, variables and parameters required to calculate potential costs. However, there is a lack of product vs. machine relational parameters hence

[92] Ford uses a proprietary system for production monitoring called POSMon, “Production Operation System Monitoring”, and is being employed in the majority of Ford’s production plants worldwide. The POSMon covers the whole production system, bringing production information (events, faults etc.) directly from the PLCs at the machine level, to the production engineers.

[93] Enterprise Resource Planning

production machine tool suppliers usually dominate this cost finalisation process and end user often find themselves unable to properly predict or properly control these costs. There is the potential to integrate cost estimation into the ALDIMS knowledge base. It is envisaged that the KB system will help provide machine tool's technical features required (standard, altered, new designed etc.) and final workstation configuration with respect to changed product, therefore, helping to provide precise estimates of costs by mapping the required changes predicted by ALDIMS to "Activity Based Costing" (ABC)[94] cost models of design, reprogram, retrofit or reconfigure at the early stages of program management. The rapid constraints evaluation provides the decision about the retrofit category of the machine. For each category, the Ford Company allocates an estimated cost for the machine. Therefore, product requirements vs machine limitations can be transformed to direct costs at the initial stages of the program management.

8.5 Concluding Remarks

The principal contribution of this research is the creation of a rapid constraint evaluation method to complement the CCE virtual engineering toolset. A new approach and a changed engineering workflow are suggested to replace current practices in order to efficiently utilise available information. The author believes that the research will have significant impacts on the assembly line design / reconfiguration. It is the first step towards transforming the well-established data-driven approach to knowledge-driven approach and from time-consuming simulation based constraint evaluation to rapid parameterised constraint evaluation of PPR inter-dependency relations.

[94] The Activity Based Costing (ABC) is a prevalent method in the automotive industry which determines cost of products, process and resources through activities performed on the domain entities.

REFERENCES

Aberdeen Group, Development Best Practices for Branded and Private Label Products in a Soft Economy, 2008.

Aletan, S., 1991, The components of a successful CIM implementation. Industrial Engineering, 23, 20-22.

Alizon Fabrice, Shooter Steve, 2005, Introduction of The Reuse Method: Retrieving Knowledge From Existing Product Designs, Proceedings of IMECE05, 2005 AMSE International Mechanical Engineering Congress and Exposition, November 5-11, 2005 – Orlando, Florida.

Alizon Fabrice, Steven B. Shooter, Timothy W. Simpson, 2006, Reuse of Manufacturing Knowledge to Facilitate Platform-Based Product Realisation, Journal of Computing and Information Science in Engineering, Vol. 6, June, 2006

Allee Verna, 2000, Knowledge Networks and Communities of Practice, OD (Organisational Development) Practitioner, Fall/Winter 2000.

Asiedu, Y and Gu, P. Product life cycle cost analysis: state of the art review. Int. J. Prod. Res, 1998, 36(4), 883–908.

Baader F, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. The Description Logic Handbook. Cambridge University Press, 2003.

Baldwin C.Y. and Clark K.B. "Design rules. Volume 1. the power of modularity", The MIT Press, Cambridge, Massachusetts, 2000.

Bedworth, D. D. et al., 1991. Computer Integrated Design and Manufacturing. New York: McGraw-Hill International Ed.

Bell, D. (1974). The coming of post-industrial society. London: Heinemann Educational Books Ltd.

Grosof Benjamin N., Ian Horrocks, Raphael Volz, and Stefan Decker. Description logic programs: Combining logic programs with description logic. In Proc. Intl. Conf. on the World Wide Web (WWW-2003), Budapest, Hungary, 2003.

Chudoba Bernd, Xiao Huang; Development of a Dedicated Aerospace Vehicle Conceptual Design Knowledge-Based System; AIAA 2006-225; 44th AIAA Aerospace Sciences Meeting and Exhibit. 9-12 January 2006, Reno, Nevada.

Bhattacharyya, S.K.; Roy, R.; Allchurch, M.; Low, M.J.; 1992, First International Conference on Intelligent Systems Engineering, 1992., (Conf. Publ. No. 360) 2002, pp 92-97.

Berger Ulrich, Ralf Kretzschmann, and A. Vargas Veronica Algebra, 2008, Intelligent Production Monitoring and Control based on Three Main Modules for Automated Manufacturing Cells in the Automotive Industry, AIP Conf. Proc. 1019, pp. 275-280.

Bernard A., Tichkiewitch S., (Eds), 2008, Methods and Tools for Effective Knowledge Lifecycle Management, Springer, Berlin, Heidelberg.

Bollinger J.G. 1998, Visionary Manufacturing Challenges for 2020, National Academy Press, Washington.

Bontis, N. and Choo, C. W., 2002, The Strategic Management of Intellectual Capital and Organizational Knowledge, Oxford University Press, New York.

Bracht, U., Masurat, T., 2005. The Digital Factory between vision and reality. Computers in Industry, Vol. 56, No. 4: 325-333

Bressan S., Goh C., Levina N., Madnick S., Shah A., Siegel M., "Context Knowledge Representation and Reasoning in the Context Interchange System", Applied Intelligence: The International Journal of Artificial Intelligence, Neutral Networks, and Complex Problem-Solving Technologies, 12(2), pp. 165-179, 2000.

Brinkop A., Laudwein N., and Maassen R., Routine Design for Mechanical Engineering. In IAAI '94 Proc. of the Sixth Annual Conference on Innovative Applications of AI, Seattle, August 1-3, 1994.

Cao R.. "Research on Manufacturing Service Matching of Internet-Based Manufacturing Based on Semantic Web." Zhejiang University Press, 2003.

Cameron I., Duff R. and Hare B. (2004). Integrated Gateways: Planning out Health and Safety Risk. Research Report 263, Glasgow Caledonian University, UK.

Chakrabarty Sugato, Rahul Chougule and Ronald M. Lesperance, 2009, Ontology-guided knowledge retrieval in an automobile assembly environment, The International Journal of Advanced Manufacturing Technology, Volume 44, Numbers 11-12, pp1237-1249.

Chau K.W., An ontology-based knowledge management system for flow and water quality modeling, Advances in Engineering Software, 38 (2007) 172–181.

Chryssolouris, G., Mavrikios, D., Papakostas, N., Mourtzis, D., Michalos, G., Georgoulias, K., 2009, Digital manufacturing: history, perspectives, and outlook, Proc. of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, Vol. 223, No. 5. ISSN 0954-4054: 451-462.

Chung C.W., Choi J.-K., Ramani K. and Patwardhan H., "Product Node Architecture: A Systematic Approach to Provide Structured Flexibility in Distributed Product Development", Concurrent Engineering Research & Application, 13 (3), pp. 219-232, 2005.

Ciocoiu, M., Gruninger, M. and Nau, D.S., Ontologies for integrating engineering applications. J. Comput. Info. Sci. Engng, 2001, 1, 12–22.

Claesson, Anders 2006, A Configurable Component Framework Supporting Platform-Based Product Development, PhD Thesis, Chalmers University of Technology, Göteborg, Sweden.

Davenport TH, Prusak L (1997) Working Knowledge: How Organizations Manage What They Know. Harvard Business School Press, Cambridge (MA).

Davenport, T.H., DeLong, D.W. and Beers, M.C. (1998) "Successful Knowledge Management Projects", Sloan Management Review, vol.39, no.2, pp.43-57

Dashchenko A., Manufacturing technologies for the machines of the future 21st century technologies, 2003, ISBN, 3540434925, Springer.

Davis, M. (2006). Semantic Wave 2006 - Executive Guide to Billion Dollar Markets. Project10X, Special Report.

de Bruijn J., H. Lausen, R. Krummenacher, A. Polleres, L. Predoiu, M. Kifer, D. Fensel, The Web Service Modeling Language WSML. WSML Deliverable D16.1, v0.2, 2005. http://www.wsmo.org/TR/d16/d16.1/v0.2/

[de Bruijn et al., 2004a] de Bruijn, J., Polleres, A., Lara, R., and Fensel, D. (2004a). OWL¡. Deliverable d20.1v0.2, WSML. Available from http://www.wsmo.org/2004/d20/d20.1/v0.2/

De Vin, Leo J. Amos H.C. Ng and Jan Oscarsson, 2004, Simulation Based Decision Support for Manufacturing System Life Cycle Management, Journal of Advanced Manufacturing Systems, Volume 3 Number 2, December 2004, pp 115-128

Dickinson Ian, 2009, Jena Ontology API from http://jena.sourceforge.net/ontology/ last accessed 19-06-2011.

Kiritsis Dimitris, A Review of Knowledge-Based Expert Systems for Process Planning. Methods and Problems, Int J Adv Manuf Technol (1995) 10:240-262.

Duineveld, A.J., Stoter, R., Weiden, M.R., Kenepa, B., and Benjamins, V.R.: WonderTools? – A comparative study of ontological engineering tools, International Journal of Human-Computer Studies, Vol. 52, 2000.

Eddy, D., J. Whitmore, K. Neville, P. Tessier, M. Dalrymple, and N. Takamoto, "Command-and-Control Decision Making: A Model for Information Operations", Proc. of 44th Annual Meeting of the Human Factors and Ergonomics Society, San Diego, CA, 2000.

Eldrandaly K., and Eldin N., "A Knowledge -based Decision Support System for Scrapers Selection and Cost Estimation" International Arab Journal of Information Technology, Vol.3, No.4, pp.339 -343, 2006.

Ericsson A. and Erixon G., "Controlling design variants: modular product platforms", ASME press, New York, 1999.

Erixon G. "Modular function deployment – a method for product modularisation", Doctoral Thesis. TRITA-MSM R-98-1, Dept. of Manufacturing Systems. Assembly Systems Div. Royal Institute of Technology. Stockholm. 1998.

Flegel Heinrich A., 2010, European RTD Framework Programme, Factories of the Future - From Economic Recovery to Sustainability, IV Conference on the FP7 in Spain,., Palacio de Congresos de Valencia, 13th April 2010

Faura, F., M.A. Sebastian, and R. Zamora, "A Decision Support System for Sheet Metal Blanking Process Parameters Selection", Journal of Material Processing Technology, 2001, pp. 371 -376.

Fensel, D. et al. (2006). Enabling Semantic Web Services: The Web Service Modeling Ontology. Springer-Verlag Ltd.,London.

Fixson, S. K., 2005. Product Architecture Assessment: A Tool to link Product, Process, and Supply Chain Design Decisions. Journal of Operations Management. 23, 3/4, 345-369.

Fixson Sebastian K, Park Jin-Kyu, The power of integrality: Linkages between product architecture, innovation, and industry structure, Research Policy (2008), Volume: 37, Issue: 8, Publisher: Elsevier, Pages: 1296-1316

Franconi E. and Tessaris S., Rules and queries with ontologies: a unified logical framework. In Workshop on Principles and Practice of Semantic Web Reasoning (PPSWR'04), St. Malo, France, 2004.

Fujita, K., & Yoshida, H. 2004. Product Variety Optimization Simultaneously Designing Module Combination and Module Attributes. Concurrent Engineering: Research and Application, 12(2): 105-118.

Gabriela Avram: At the Crossroads of Knowledge Management with Social Software. ECKM 2005: 49-58

Germani, M., & Mandorli, F. 2004. Self-configuring components approach to product variant development. Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 18: 41-54.

Goh C.H., S. Bressan, S. Madnick, M. Siegel, "Context Interchange: New Features and Formalisms for the Intelligent Integration of Information", ACM Trans. on Information Systems (TOIS), 13(3), 270-293, July 1999.

Gourley David & Totty Brian, HTTP: The Definitive Guide, O'Reilly Media, Inc. 2002

Grabot, B., J. Blanc, and C. Binda, "A Decision Support System for Production Activity Control", Decision Support Systems, Vol. 16, No. 2, 1996, pp. 87-101.

Gruninger M., 2004, "Ontology of the Process Specification Language." http://www.mel.nist.gov/psl/psl-ontology/

Gruber T. R., "A Translation Approach to Portable Ontology Specifications", Knowledge Acquisition, vol. 5(2), 1993, pp. 199-220.

Gruber Tom. Towards principles for the design of ontologies used for knowledge sharing. Int J Hum Comput Stud 1995;43(5–6):907–28.

Gruninger M., and Menzal C.. "Process Specification Language (PSL): Theory and Application.". AI magazine, vol. 24, no. 3, pp. 63-74, Fall 2003.

Haegele Martin, Thomas Skordas, Stefan Sagert, Rainer Bischoff, Torgny Brogårdh, Manfred Dresselhaus, 2005, White paper - Sectoral Report on Industrial Robot Automation, Network of Excellence, Information Society Technologies, July, 2005.

Hahn, A. (2007). Using Ontologies to Model and Understand Product Development. IBIS –Interoperability in Business Information Systems, 5, 21-39.

Haq I., (2009), 'Innovative configurable and coolaborative approach to automation systems engineering for automotive powertrain assembly', PhD thesis, Loughborough University, UK.

Harrison, R.; Monfared, R.P.; Les Lee; 2009, Business driven engineering for powertrain industry, IEEE Conference on Emerging Technologies & Factory Automation, 2009. pp 1-4, ETFA 2009.

Harrison, R., A.W.Colombo, A.A.West and S.M.Lee, 2006, Reconfigurable modular automation systems for automotive power-train manufacture. International Journal of Flexible Manufacturing Systems, 18(3): p. 175-190.

Hayes-Roth, F. and Jacobstein, N. (1994). The State of Knowledge-Based Systems, Communications of the ACM, vol. 37, no. 3, March, pp. 27-39, ACM Press.

Hendriks PHJ. Many rivers to cross: from ICT to knowledge management systems. Journal of Inform Technol 2001; 16(2):57–72.

Hepp, M. (2007). Ontologies: State of the Art, Business Potential, and Grand Challenges. Ontology Management: Semantic Web, Semantic Web Services, and Business Applications, Springer, 3-22.

Lausen Holger, Francisco Martin-Recuerda, Jos de-Bruijn, and Michael Stollberg (University of Innsbruck), 2005, A Conceptual and Formal Framework for Semantic Web Services (v1), EU-IST Network of Excellence (NoE) IST-2004-507482 KWEB Deliverable D2.4.5v1.

Horrocks I., P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean. SWRL: A Semantic Web Rule Language Combining OWL and RuleML. W3C Member Submission, 21 May 2004. Available from http://www.w3.org/Submission/SWRL/.

Horrocks I., P. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M.Dean. SWRL: A semantic web rule language combining owl and Rule-ML. DARPA DAML Program, 2003. http://www.w3.org/Submission/2004/SUBM-SWRL-20040521/.

Hsiao Shih-Wen, Huang H.C., 2002, A neural network based approach for product form design, Design Studies, Vol. 23: 67-84

Hu, S.J., 2005, Paradigms of manufacturing - A panel discussion, 3rd Conference on Reconfigurable Manufacturing, Ann Arbor, Michigan, USA.

Huber GP. Transfer of knowledge in knowledge management systems: unexplored issues and suggested studies. Eur J Inform Syst 2001;10(2):72–9.

IMTR (1999), Integrated manufacturing technology roadmapping (IMTR) project - information systems for the manufacturing enterprise, http://imti21.org/

Ishi, T., Driving High Quality and Low Costs in a Rapidly-Changing World, Gensym User Society Worldwide Meeting, Nice, France, October 6-8, 2004, (2004).

Jammes F. and Smit H., "Service-oriented architectures for devices - the SIRENA view," 3rd IEEE International Conference on Industrial Informatics (INDIN), 2005, pp. 140-147.

Jarvenpaa, E.; Luostarinen, P.; Lanz, M.; Tuokko, R.; 2011, Presenting capabilities of resources and resource combinations to support production system adaptation , IEEE International Symposium on Assembly and Manufacturing (ISAM), 2011, TUT, Finland.

Lanz Minna, 2010, Logical and Semantic Foundations of Knowledge Representations for Assembly and Manufacturing Processes, PhD Thesis, Tampere University of Technology, Finland.

Jarvenpaa E, Lanz M, Tuokko R, Studying the information sources and flows in a company supporting the developement of new intelligent systems, In proceedings of Flexible Automation and Intelligent Manufacturing, 2010.

Jetter, A., Kraaijenbrink, J., Schroeder, H-H, Wijnhoven, F. (eds.) (2005): Knowledge integration: The practice of knowledge management in small and medium sized enterprises. Springer, Physica-Verlag, Heidelberg.

Klampfl, E., Gusikhin, O., and Rossi, G., e-Workcell Application in a mixed-model automotive assembly environment, Proceedings of 2004 JUSFA, 2001 Japan - USA Symposium of Flexible Automation, Denver, Colorado, July 19-21, 2004.

Kifer Michael, Lausen Georg, and Wu James. Logical foundations of object-oriented and frame-based languages. JACM, 42(4):741–843, 1995.

Kitamura, Y., Koji, Y. & Mizoguchi, R. 2006. “An ontological model of device function: industrial deployment and lessons learned”. Journal of Applied Ontology, Vol. 1, No. 3-4, pp. 237-262.

Klampfl, E., Gusikhin, O., and Rossi, G., Optimization of workcell layouts in a mixed-model assembly line environment, Int J Flex Manuf Syst, vol. 17, 2006, pp. 277-299.

Konigsberger H.K. and De Bruyn F.W.G.M. (1990), Prolog from the Beginning, McGraw-Hill Book Company (UK) Limited, London, England.

Koren Y., Heisel U., Jovane F., Moriwaki T., Pritschow G., Ulsoy G., Van Brussel H., 1999, Reconfigurable manufacturing systems, CIRP Annals 48(2): 527-540.

Koren, Y. and Ulsoy, G,: Reconfigurable Manufacturing System Having a Method for Changing its Production Capacity. US patent # 6,349,237; issue date: 2/19/2002.

KPMG's Global Auto Executive Survey, 2010.

Kühn, W., 2006, Digital Factory – Integration of Simulation Enhancing the Product and Production Process towards Operative Control and Optimisation, International Journal of Simulation, Vol. 7, No 7, ISSN 1473-8031: 27-39.

Küsters, K. (2006). Teilautomatisierte Überführung der Produktstruktur in eine lastfallabhängige Modellstruktur mittels Ontologien. Diplomarbeit Rheinisch-Westfälische Technische Hochschule Aachen.

Kwan Hee Han and Namchul Do, An object-oriented conceptual model of a collaborative product development management (CPDM) system, The International Journal of Advanced Manufacturing Technology, Volume 28, Numbers 7-8, 827-838.

Lai H. Y. Steven, 1993, KBDA - A Knowledge Based Design for Assembly System, Proceedings of the 15th Annual Conference on Computers and Industrial Engineering, Computers and Industrial Engineering Vol. 25, Nos 1-4, pp. 585-588.

Landers, R., Min, B.K., and Koren, Y.: Reconfigurable Machine Tools. CIRP Annals, Vol. 49, No. 1, pp. 269-274, July 2001.

La Rocca G., Krakers L., M.J.L. van Tooren; Development of an ICAD Generative Model for Blended Body Aircraft Design; AIAA 2002-5447; 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization. 4-6 September 2002, Atlanta, Georgia.

Lastra Jose L. Martinez, Delamer Ivan M., 2006, Ontology Modeling of Assembly Processes and Systems using Semantic Web Services, IEEE International Conference on Industrial Informatics, Singapore, pp611-617.

Leger, J.-B., & Morel, G. 2001. Integration of maintenance in the enterprise: towards an enterprise modeling-based framework compliant with proactive maintenance strategy. Production Planning & Control, 12(2): 176-187.

Li, R. K. and Hwang, C. L. A framework for automatic DFA system development. Computers Ind. Engng, 1992, 22(4), 403–413.

Liebowitz J. Knowledge management and its link to artificial intelligence. Expert Syst Appl 2001; 20(1):1–6.

Lin, H.K. and Harding, J.A.(2003) An ontology driven manufacturing system engineering moderator for global virtual enterprise teams. Proceedings of the 1st International Conference on Manufacturing Research (ICMR),Glasgow, UK, 2003, pp. 365–370.

Lloyd J. W.. Foundations of Logic Programming (2nd edition). Springer-Verlag, 1987.

Lohse 2006, Towards an ontology framework for the integrated design of modular assembly systems, PhD Thesis, University of Nottingham

Maier, R. (2004) Knowledge Management Systems, Springer, pp 361-368, 503-512.

Manufuture (2006a) – Strategic Research Agenda – Assuring the future of manufacturing in Europe: Report of the high Level Group, September 2006, Available from: http://www.manufuture.org/ [Accessed: 14.09.2010]

ManuFuture (2006b) "Strategic Research Agenda: assuring the future of manufacturing in Europe", June 2006

ManVis (2005a) "ManVis Report 6 – Policy Summary and Recommendations" Manufacturing Visions, [Website] Available from: http://forera.jrc.es/documents/PolicyPaper_final.pdf [Accessed: 14.09.2010]

Marcellus D.H. (1989), Expert Systems Programming in Turbo Prolog, Prentice-Hall Inc, New Jersey.

Marshall, R. & P.G. Leaney, 1995. Modularisation as a Means of Product and Process Integration. Advances in Manufacturing Technology IX. Proceedings of the 11th National Conference on Manufacturing Research. London: Taylor & Francis, 1995, pp. 129-133.

Masahiro Hori, Jérôme Euzenat, Peter F. Patel-Schneider. W3C Note 11 June 2003, available from http://www.w3.org/2004/OWL/ last accessed 20-06-2011

Mcevily, S., and Chakravarthy, B., 2002, "The Persistence of Knowledge-Based Advantage: An Empirical Test for Product Performance and Technical Knowledge," Strategic Manage. J., 23(1), pp. 285–305.

Mehrabi, M.G., Ulsoy A.G., Koren Y., Heytler P., 2002, Trends and perspectives in flexible and reconfigurable manufacturing systems, Journal of Intelligent Manufacturing, Vol. 13: 135-146

Mehrabi, M.G., Ulsoy, A. G., Koren, Y., 2000, Reconfigurable manufacturing systems: key to future manufacturing, Journal of Intelligent Manufacturing, 11: 413-419.

Milani A. A., Hamedi Mohsen, A Knowledge-based System for Selecting Fastening Tools in Automobile Assembly Lines, IEEE International Conference on Industrial Technology, 2008. ICIT 2008.

Miles J. and Moore C. (1994) Practical knowledge based systems for conceptual design, pp 35-53, Springer-Verlag, Berlin, Germany.

Ming C., Lin L., and Dong J.. "Intelligent Service Matching in Internet-Based Manufacturing." CIMS, vol.9, no. 7, pp. 589-594, March 2003.

Miresco E.T. and Pomerol J.C. (1995). A knowledge-based decision support system for construction project management, Proceedings of the Sixth International Conference on Computing in Civil and Building Engineering, No.2, pp. 1501-1507.

Molina, A.; Rodriguez, C.; Ahuett, H.; Cortés, J.; Ramírez, M.; Jiménez, G.; Martinez, S., Next-generation manufacturing systems: key research issues in developing and integrating reconfigurable and intelligent machines, International Journal of Computer Integrated Manufacturing, Volume 18, Number 7, October-November 2005 , pp. 525-536(12)

Molloy, F., H. Yang, and J. Browne. "Feature-based modeling in design for Assembly," International Journal of Computer Integrated Manufacturing, Vol.6, No. 1-2, Jan - April 1993, pp. 119-125.

Moula M (1993) A knowledge based system to assist in the selection of appropriate geotechnical field tests, PhD Thesis, School of Engineering and Computer Science, University of Durham.

Nawijn M., MJ.L. van Tooren, J.P.T.J. Berends, P. Arendsen; Automated Finite Element Analysis in a Knowledge Based Engineering Environment; AIAA 2006-947; 44th AIAA Aerospace Sciences Meeting and Exhibit. 9-12 January 2006, Reno, Nevada.

Neches R., Fikes R., Finin T., Gruber T., Patil R., Senator T. and Swartout W., "Enabling Technology For Knowledge Sharing", AI Magazine, vol. 12(3), 1991, pp. 36-56.

Nonaka I. And D. Teece (2001) Managing Industrial Knowledge: creation, transfer and utilisation. London; Thousand Oaks, Calif., SAGE.

Nonaka, I. & Takeuchi, H. (1995). The knowledge-creating company. New York: Oxford University Press

Noy, N.F. and Klein, M.C.A. (2004), Ontology evolution: not the same as schema evolution, Knowledge and Information Systems, pp. 428–440, 2004.

Obitko, M., and Marik,V.. "Adding OWL Semantics to Ontologies Used in Multi-agent Systems for Manufacturing." Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science), 2003, 2744: pp.189-200.

O'Brien J.J. (1998). Construction Change Orders. McGraw Hill, New York.

Oliver Andy, 1994, A knowledge based system for the interpretation of site investigation information, Doctoral thesis, Durham University.

OMG (2005), On the road, Available online at: http://www.omg.org [Accessed on 20 March 2011]

Owen T. Assembly with robots, 1985 (Kogan Page, London).

Pal, K., & Palmer, O. (2000). A decision support system for business acquisitions. Decision Support Systems, 27, 411–429.

Pundt H, Bishr Y. Domain ontologies for data sharing – an example from environmental monitoring using field GIS. Comput Geosci 2002; 28(1):95–102.

Quinn JB (1992) Intelligent Enterprise: A Knowledge and Service Based Paradigm for Industry. The Free Press, New York.

Ray S. and Jones A., 2006, Manufacturing Interoperability, Journal of intelligent manufacturing (17), pp 681-688.

Raza Muhammad Baqar, Robert Harrison, Tom Kirkham, 2009a, "Knowledge Based Product Lifecycle Management System", International Journal of Computing Architecture, Volume 4, Issue 2.

Raza Muhammad Baqar, Robert Harrison, Tom Kirkham, 2009b, "Improving Manufacturing Efficiency at Ford using Product Centred Knowledge Management" in the Proceedings of the IEEE, 4th International Conference on ICDIM, Michigan, USA, November, 2009.

Raza Muhammad Baqar, Robert Harrison, Tom Kirkham, Q Reul, 2009c, "Knowledge Based Flexible and Integrated PLM System at Ford", in IJISM , Vol. 1, No.1, March, 2011, pp 8-16.

Raza Muhammad Baqar, Robert Harrison, 2011a, "Intelligent Integration of Product Design to Production Devices", Proceedings of the 10th International Conference on Information and Knowledge Engineering (ICIKE, WORLDCOMP 2011,), Las Vegas, Nevada, USA, July, 2011.

Raza Muhammad Baqar, Robert Harrison, 2011b, “Ontological Knowledge Based System for Product, Process and Resource Relationships in Automotive Industry”, Proceedings of the 1st International Conference on Ontology and Semantic Web for Manufacturing (OSEMA, 2011), collocated with the 8th Extended Semantic Web Conference (ESWC), Greece, May, 2011.

Raza Muhammad Baqar, Robert Harrison, 2011c, “Information Modeling and Knowledge Management Approach to Reconfiguring Manufacturing Enterprises”, IJAIT, World Scientific, Vol. 20, No. 3, June, 2011.

Raza Muhammad Baqar, Robert Harrison, 2011d, “Design, Development & Implementation of Ontological Knowledge Based System for Automotive Assembly Lines”, International Journal of Data Mining & Knowledge Management Process (IJDKP) Vol. 1, No.5, September, 2011.

Raza Muhammad Baqar, Robert Harrison, Tom Kirkham, 2011e, “Knowledge Based Services for Devices in Automation”, - Proceedings of the International Joint Conference on Knowledge Engineering and Ontology Development, KEOD, IC3K, Paris, France, October, 2011.

Raza Muhammad Baqar, Robert Harrison, 2012a, “Amalgamation of Ontology and Knowledge Engineering in Automotive Industry”, International Journal of Advanced Computer Science and Applications, IJACSA, Vol. 3, No. 1, January, 2012.

Raza Muhammad Baqar, Robert Harrison, Tom Kirkham, 2012b, “Embracing Knowledge Driven Devices: Service Oriented Standardisation of Production Management in Automated Lines”, International Journal of Service and Computing Oriented Manufacturing, IJSCOM, Inderscience Publishers, March, 2012.

Remus U, Schub S (2003) A blueprint for the implementation of process-oriented knowledge management. Knowledge and Process Management, 10 (4): 237–253.

Krummenacher Reto, Holger Lausen, Thomas Strang, On the Modeling of Context-Rules with WSML, Digital Enterprise Research Institute (DERI), University of Innsbruck, Austria, 2007.

Roche, C.,:Corporate ontologies and concurrent engineering, Journal of Materials Processing Technology, Vol.107, 2000, p.187-193.

Roy J, 2007, A knowledge-centric view of situation analysis support systems, Technical Report, DRDC Valcartier TR 2005-419, Defence R&D Canada

Klein Ruediger, 2005, Towards an integration of engineering knowledge management and knowledge based engineering, DaimlerChrysler Research and Technology, Knowledge Based Engineering Group, Berlin, Germany.

Rychtyckyj, N., Intelligent Systems for Manufacturing at Ford Motor Company, IEEE Intelligent Systems, vol. 22, no. 1, pp. 16-19, Jan/Feb, 2007.

Saha G. K., "Web Ontology Language (OWL) and Semantic Web", Ubiquity vol. 8(35), 2007, pp. 1-24.

Sánchez D, Moreno A (2008). Learning non-taxonomic relationships from web documents for domain ontology construction. Data Knowl.Eng., 64: 600–623.

Sandberg Marcus, Larsson Tobias, Automating redesign of sheet-metal parts in automotive industry using KBE and CBR, Proceedings of IDETC/CIE 2006, ASME 2006 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference September 10-13, 2006, Philadelphia, Pennsylvania, USA.

Schlenoff, C., Gruninger, M., Tissot, F., Valois, J., Lubell, J. and Lee, J.(2000) The process specification language (PSL) overview and version 1.0 specification. NISTIR 6459, Gaithersburg, MD, 2000.

Schnitger Monica, “Digital Simulation to Meet Today’s Product Development Challenges“, Daratech White Paper, December 2003.

Schreiber, G., Akkermans, H., Anjewierden, A., de Hoog, R., Shadbolt, N., de Velde, W.V. and Wielinga, B. (1999), Knowledge Engineering and Management: The CommonKADS Methodology, MIT Press, Massachusetts.

Sharma A., Collaborative product innovation: integrating elements of CPI via PLM framework, Computer Aided Design, 2005, 37 (13): p. 1425-1434.

Shehab, E. M and Abdalla, H. S. A design to cost system for innovative product development. Proc. Instn Mech Engrs, Part B: J. Engineering Manufacture, 2002, 216(B7), 999–1020.

Shehab E. M., Abdalla H. S.. 2006, A cost-effective knowledge-based reasoning system for design for automation, Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, Volume 220, Number 5 / 2006 pp 729-743.

Slotwinski John A., Tilove Robert B., Smart Assembly, Industry needs and challenges. National Institute of Standards and Technology. Published in: PerMIS '07 Proceedings of the 2007 Workshop on Performance Metrics for Intelligent Systems, ACM New York, NY, USA, 2007. http://smartassembly.wikispaces.com/

Sly David, Ph.D., Research Associate 2004, Manufacturing Process Management (MPM), CPDA: Collaborative Product Development Associates, White paper. Available from: http://www.proplanner.com/documents/filelibrary/documents/papers_case_studies/MPM_Whitepaper__Tech_Trend_PDF_CDF4B29897EE8.pdf Last accessed 16-06-2011.

Sowa J. F., Knowledge Representation, Brooks/Cole, an imprint of Thomson Learning, 2000.

Staab, S. (2002). Wissensmanagement mit Ontologien und Metadaten Habilitation Thesis, University of Karlsruhe.

Staab, S., H.-P. Schnurr, R. Studer, Y. Sure: Knowledge Processes and Ontologies. IEEE Intelligent Systems. 16(1), January/February 2001. Special Issue on Knowledge Management.

Stecke, K.E., 2005, Flexibility is the future of reconfigurability, Paradigms of Manufacturing - A Panel Discussion, 3rd Conference on Reconfigurable Manufacturing, Ann Arbor, Michigan, USA

Stegmüller, H. (2003). Zeit fürs Wesentliche - Audi erprobt semantische Technologien. Digital Engineering Magazin, 4, 44-45.

Studer, R., Benjamins, R.V., and Fensel, D(1998), Knowledge Engineering: Principles and Methods. Data & Knowledge Engineering, 25: p. 161-197.

Sudin Mohd Nizam, Harun Mohd Ruzi, Ahmad Tajudin Hamzah and Shamsul Anuar, A prototype of KBS for material selection in bumper beam design, 2007, Suranaree J. Sci. Technol. 14(3):215-222.

Sundin Erik, 2004, Product and Process Design for Successful Remanufacturing, PhD dissertation, Production Systems, Department of Mechanical Engineering, Linköpings Universitet, Linköping, Sweden

Sutcliffe A., 1988, Human Computer Interface Design MacMillan Education Ltd. London.

Syldatke, T. et al. (2007). How Ontologies and Rules Help to Advance Automobile Development. Advances in Rule Interchange and Applications. International Symposium, RuleML 2007, Orlando, Florida, October 25-26, 2007, Proceedings, Invited Paper, 1-6.

Tay FEH, Ming C (2001) A shared multi-media design environment for concurrent engineering over the internet. J Concurr Eng: Res Appl 9(1):55–63.

Tiwana A, Ramesh B. A design knowledge management system to support collaborative information product evaluation. Decision Support System 2001; 31(2):241–62.

TRANSPARENCY Project, Adaptive Business Collaboration by Progressive Knowledge Sharing and Engineering, 2011. Available from www.transparency-project.com

Turban, R. S. E., Aronson, J. E. & Liang, T. P. (2007). 'Decision Support and Business Intelligence Systems (8th International Edition),' Pearson International Edition.

Tursi A., Panetto H., Morel G., Dassisti M. (2009). Ontological approach for Products-Centric Information System, Interoperability in Networked Manufacturing Enterprises. IFAC Annual Reviews in Control. 33/1, April, Elsevier.

Ulrich Karl, “The role of product architecture in the manufacturing firm”, Research Policy 24 (1995).

Upton, D. M., & McAfee, A. P. 2000. A path-based approach to information technology in manufacturing. International Journal of Technology Management, 20(3/4): 354-372.

Uschold Mike Gruninger Michael, 1996, Ontologies, Principles Methods and Applications, Knowledge Engineering Review, Volume 11 Number 2 June 1996

US National Research Council (1998). Visionary Manufacturing Challenges for 2020. Washington, D.C., National Academy Press.

VanVeen, E.A. (1992) Modelling product structures by generic bills-of-material, Elsevier Science Publishers, Amsterdam.

VDW. "The German Machine Tool Industry in 1999." Franfurt am Main: Verein Deutscher Werkzeugmaschinenfabriken e.V. (VDW), 1999.

Venkatachalam, A. R., Mellichamp, J. M., and Miller, D. M. A knowledge-based approach to design for manufacturability. J. Intell. Mfg, 1993, 41, 355–366.

Wang Bin, Wang Ping, Liu Defang, Research on Knowledge-Based Product Rapid Design System, IEEE, 2010, International Conference on Networking and Digital Society, 2010.

Wang Guoxin, Yan Yan, Lichen Hu, Xiang Zhang, Lu Wang, Construction of knowledge based engineering platform for armored vehicle, IEEE International Conference on Industrial Engineering and Engineering Management, 2009. IEEM 2009.

Watanabe, C., & Ane, B. K. 2004. Constructing a virtuous cycle of manufacturing agility: concurrent roles of modularity in improving agility and reducing lead time. Technovation, 24: 573-583.

Wie Yui, Gary W. Fischer, and Jose L. T. Santos, “Concurrent Engineering Design Environment for Generative Process Planning Using Knowledge-based Decisions,” 1990 ASME Design Technical Conference, Chicago, Sept. 1990, pp. 35-45.

Wijnhoven, Fons. (2005), Knowledge management: More than a buzz word. In: Jetter, A., Kraaijenbrink, J., Schroeder, H-H, Wijnhoven, F. (eds.) (2005): Knowledge integration. Springer, Physic Verlag, Berlin, pp. 1–16.

Xuemei Huang, 2007, Implementing Manufacturing Reconfiguration Methodology in Multi Agent System of Reconfigurable Assembly Line, Proceedings of the 2007 IEEE International Conference on Mechatronics and Automation, August 5 - 8, 2007, Harbin, China.

Xuemei Huang, Harbin Eng. Univ., Harbin., 2007, Implementing Manufacturing Reconfiguration Methodology in Multi Agent System of Reconfigurable Assembly Line International Conference on Mechatronics and Automation, 2007.

Yongqian Zheng, A knowledge based production line development management system, International Conference on Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007.

Zatarain, M.; Lejardi, E.; Egana, F: Modular Synthesis of Machine Tools, Annals of the CIRP, 37/1, 1998, 333-336.

Zha X. F., Du, H. J., and Qiu, J. H. Knowledge-based approach and system for assembly oriented design. Part I: the approach. Engng Applic. Artif. Intell., 2001, 14(1), 61–75.

Zhang W. Y., Yin J. W., Lin L. F. & Zhu T. H. (2009): Towards a general ontology of multidisciplinary collaborative design for Semantic Web applications, International Journal of Computer Integrated Manufacturing, 22:12, 1144-1153.

Zhou Hong, 2011, A Knowledge Based Reengineering Approach via Ontology and Description Logic, Ph.D Thesis, De Montfort University, UK.

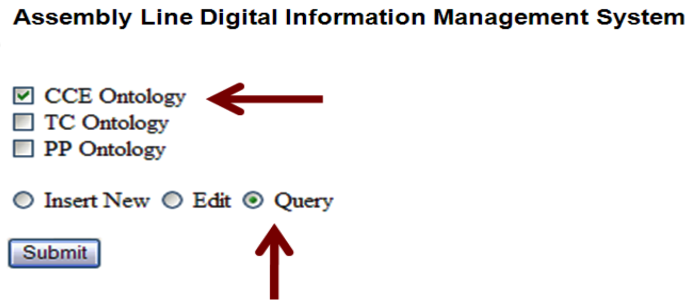
Appendix 1.

The ALDIMS User Interface

Interface through Web Page

The Search Module

The search module helps the users to make queries regarding PPR domains. The queries may be made directly to the database or can be mediated to get all the relevant concepts from different databases as required. With the single query, the data from multiple sources can be selected. For example, if the user wants to retrieve workstation, the user can select CCE, TC and PD to retrieve all workstations which have different terminology / vocabulary for workstations in different applications.



Assembly Line Digital Information Management System

☑ CCE Ontology
□ TC Ontology
□ PP Ontology

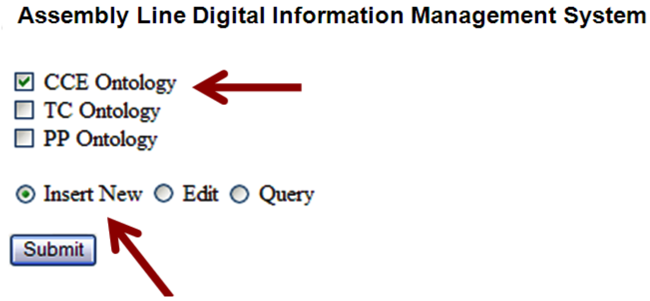
○ Insert New ○ Edit ● Query

Figure 91: The Search module form

To find a specific concept from a certain application, the respective application can be checked and to select query, the radio button beside query can be ticked, as shown in Figure 91.

The Knowledge Generation Module

The knowledge generation module is used to add new knowledge into the KB system. This can be for any of the 03 domains. As the user selects ‘Insert New’, the system provided options with template based structure to be added into the KB system. An example of adding a new machine (System) is shown in Figure 92.



Assembly Line Digital Information Management System

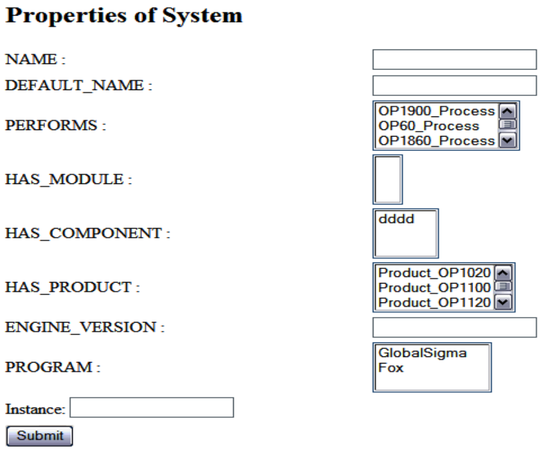
☑ CCE Ontology
□ TC Ontology
□ PP Ontology

● Insert New ○ Edit ○ Query

Submit

Figure 92: The knowledge generation module form

With the template based, user will always enter structured and complete information of the concepts.



Properties of System

NAME :	
DEFAULT_NAME :	
PERFORMS :	OP1900_Process OP60_Process OP1860_Process
HAS_MODULE :	
HAS_COMPONENT :	dddd
HAS_PRODUCT :	Product_OP1020 Product_OP1100 Product_OP1120
ENGINE_VERSION :	
PROGRAM :	GlobalSigma Fox

Instance:

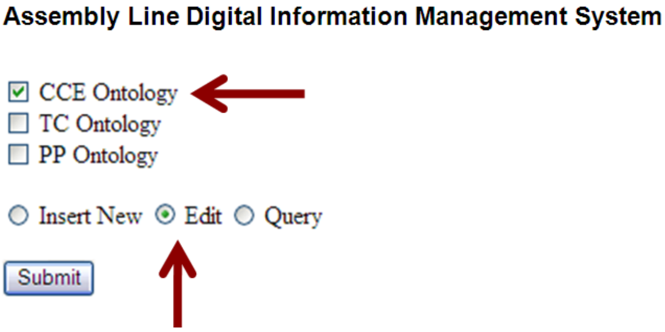
Submit

Figure 93: Adding a new machine into the KB system

The Figure 93 shows concepts and properties and user is required to select the respective concept and properties to add new knowledge in the ontological knowledge based system.

Edit Module

This module is used to edit any already added knowledge. This module can only be accessed by system developers with special rights.



Assembly Line Digital Information Management System

☑ CCE Ontology
□ TC Ontology
□ PP Ontology

○ Insert New ● Edit ○ Query

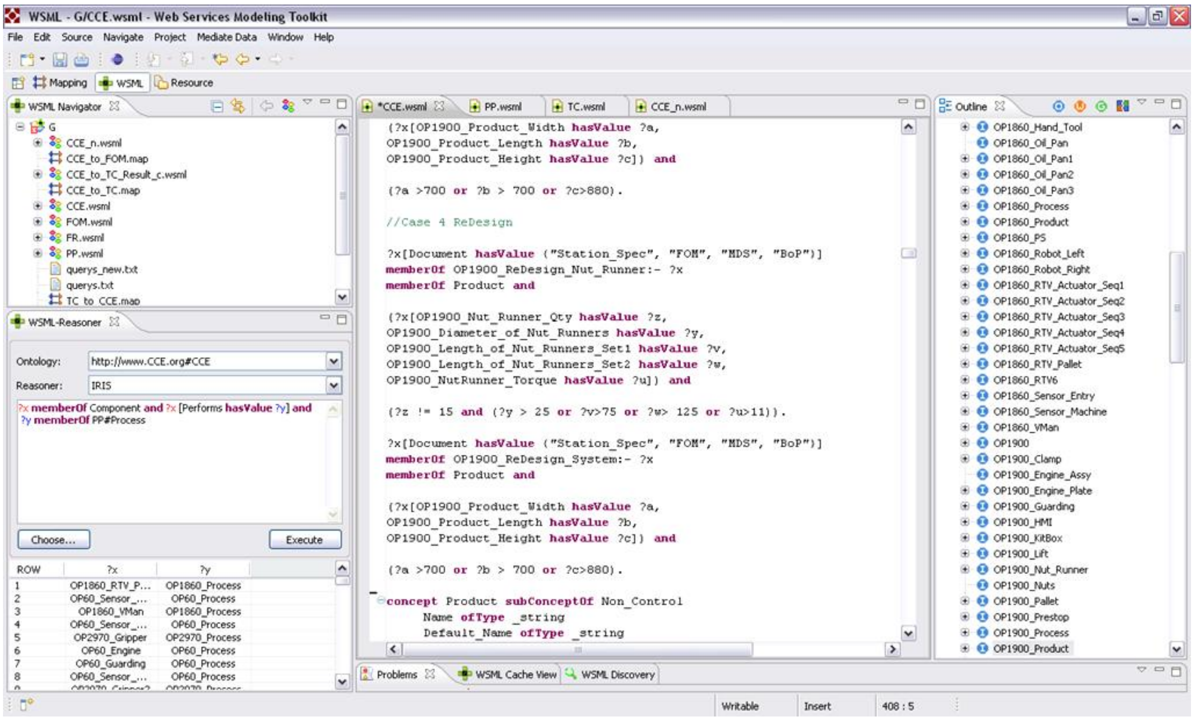
Submit

Figure 94: The Edit module form in the KB system

Figure 94 depicts the simple steps to edit any already existing knowledge in the system. As described before, the user needs to select the respective ontology and edit option to carry out editing.

Direct WSMT Interface

The previous section has described the searching, editing or inserting options through web based form however the option to access the ontologies directly through the IDE environment is also available. The foundational architecture of the KB system is built upon ontology and the language selected for the ontology is the WSML. WSML provides an IDE environment for building ontologies through an editor called WSMT which is shown in Figure 95 below:



```
(?x[OP1900_Product_Width hasValue ?a,
OP1900_Product_Length hasValue ?b,
OP1900_Product_Height hasValue ?c]) and

(?a >700 or ?b > 700 or ?c>880).

//Case 4 ReDesign

?x[Document hasValue ("Station_Spec", "FOM", "MDS", "BoP")]
memberOf OP1900_ReDesign_Nut_Runner:- ?x
memberOf Product and

(?x[OP1900_Nut_Runner_Qty hasValue ?z,
OP1900_Diameter_of_Nut_Runners hasValue ?y,
OP1900_Length_of_Nut_Runners_Set1 hasValue ?v,
OP1900_Length_of_Nut_Runners_Set2 hasValue ?w,
OP1900_NutRunner_Torque hasValue ?u]) and

(?z != 15 and (?y > 25 or ?v>75 or ?w> 125 or ?u>11)).

?x[Document hasValue ("Station_Spec", "FOM", "MDS", "BoP")]
memberOf OP1900_ReDesign_System:- ?x
memberOf Product and

(?x[OP1900_Product_Width hasValue ?a,
OP1900_Product_Length hasValue ?b,
OP1900_Product_Height hasValue ?c]) and

(?a >700 or ?b > 700 or ?c>880).

concept Product subConceptOf Non_Control
     Name ofType _string
     Default_Name ofType _string
```

WSML-Reasoner

Ontology: http://www.CCE.org#CCE

Reasoner: IRIS

```
?x memberOf Component and ?x [Performs hasValue ?y] and
?y memberOf PP#Process
```

Choose... Execute

ROW	?x	?y
1	OP1860_RTV_P...	OP1860_Process
2	OP60_Sensor_...	OP60_Process
3	OP1860_VMan	OP1860_Process
4	OP60_Sensor_...	OP60_Process
5	OP2970_Gripper	OP2970_Process
6	OP60_Engine	OP60_Process
7	OP60_Guarding	OP60_Process
8	OP60_Sensor_...	OP60_Process

Figure 95: Accessing the KB system directly through WSMT editor

The WSMT editor is an IDE based on the WSML language. The WSMT IDE editor is similar to Eclipse Java IDE in many respects. A detailed description is provided in the form of a tutorial on the WSML project websites[95], describing complete guidance from installation to creating first

[95] http://www.wsmo.org/TR/d9/d9.1/v0.2/20050425/ ; http://www.wsmo.org/TR/d9/d9.2/v0.1/20050321/

ontology, however, a brief introduction is provided here. WSMT editor is a simple and fully featured ontology development IDE. The WSMT version 1.4 is used for development of ontologies in the current research which can be downloaded free of charge from the internet[96]. The WSMT requires Java 5.0 or higher installed as the default JRE on the system. One of the features of the editor is a simple GUI with self explanatory instructions. It has 06 major sections, discussed briefly in the next sections.

WSMT Menu Bar

The main menu bar is consistent with the usual window based applications with almost the same options as provided in windows applications. There are different options for opening a requisite perspective to build required application. This is provided by 'Open Perspective' icon. The 'Open Perspective' icon option provides access to one of the 03 main perspectives defined in the WSMT i.e.

1. WSML Perspective
2. Resource Perspective
3. Mapping Perspective

Currently WSML Perspective is used in the development of the KB system, as shown in Figure 44, however, Mapping Perspective will also be used which is described in chapter 6.

WSML Navigator Window

Once the WSML perspective is selected, the user can access the 'WSML Navigator' window as shown in Figure 96 below:

[96] http://sourceforge.net/news/?group_id=154080

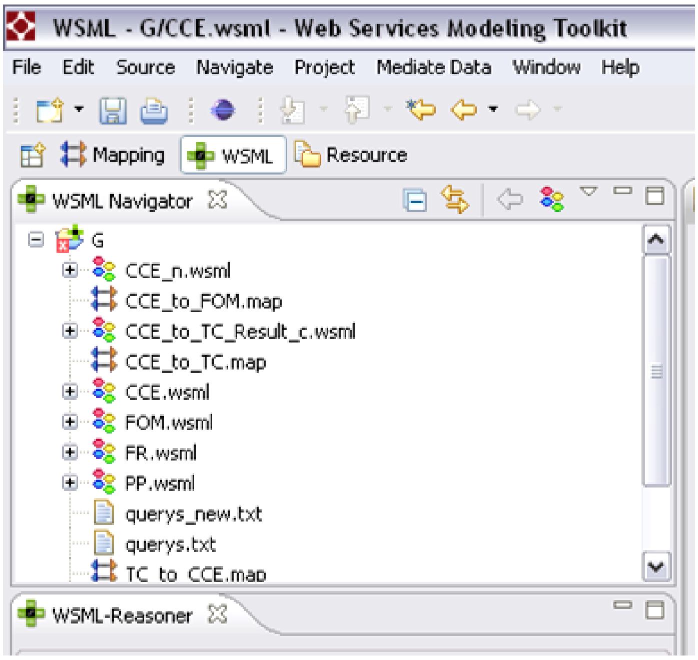
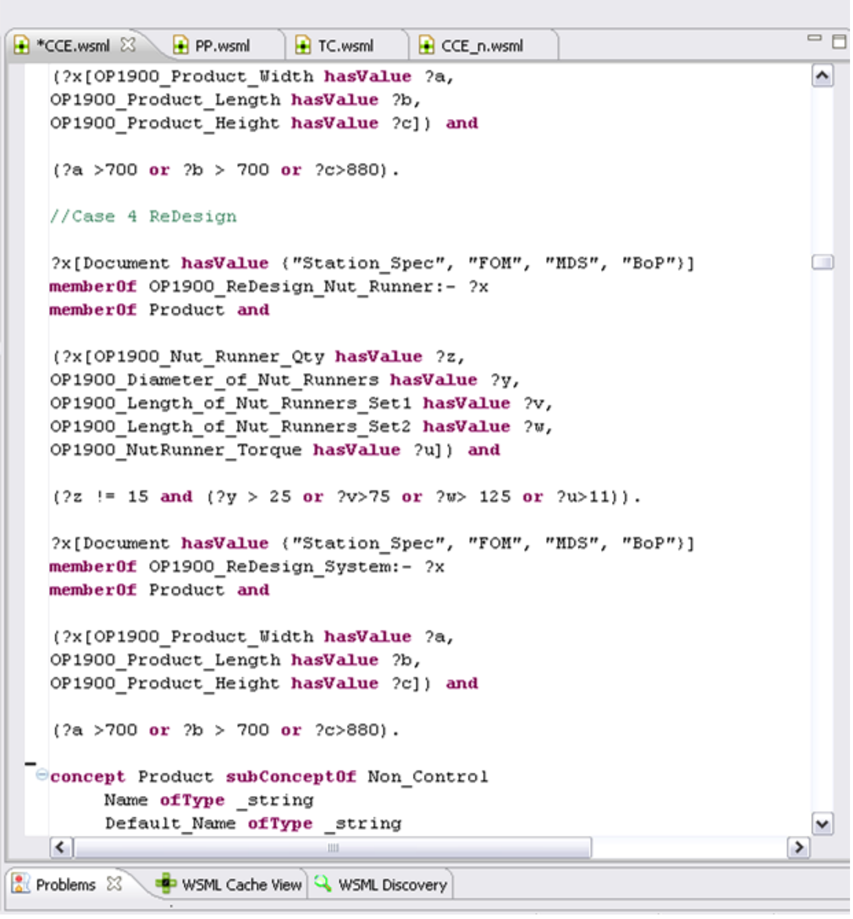


Figure 96: WSML navigation window

The WSML Navigator window allows the user to add new projects, ontologies and mapping in the projects similar to the Eclipse Java IDE.

Text Editing Area

The WSMT text editor is the dedicated input place for development of ontology. All the concepts, properties, relations, rules and axioms are defined in this area as shown in Figure 97:



```
(?x[OP1900_Product_Width hasValue ?a,
OP1900_Product_Length hasValue ?b,
OP1900_Product_Height hasValue ?c]) and

(?a >700 or ?b > 700 or ?c>880).

//Case 4 ReDesign

?x[Document hasValue {"Station_Spec", "FOM", "MDS", "BoP"}]
memberOf OP1900_ReDesign_Nut_Runner:- ?x
memberOf Product and

(?x[OP1900_Nut_Runner_Qty hasValue ?z,
OP1900_Diameter_of_Nut_Runners hasValue ?y,
OP1900_Length_of_Nut_Runners_Set1 hasValue ?v,
OP1900_Length_of_Nut_Runners_Set2 hasValue ?w,
OP1900_NutRunner_Torque hasValue ?u]) and

(?z != 15 and (?y > 25 or ?v>75 or ?w> 125 or ?u>11)).

?x[Document hasValue {"Station_Spec", "FOM", "MDS", "BoP"}]
memberOf OP1900_ReDesign_System:- ?x
memberOf Product and

(?x[OP1900_Product_Width hasValue ?a,
OP1900_Product_Length hasValue ?b,
OP1900_Product_Height hasValue ?c]) and

(?a >700 or ?b > 700 or ?c>880).

concept Product subConceptOf Non_Control
     Name ofType _string
     Default_Name ofType _string
```

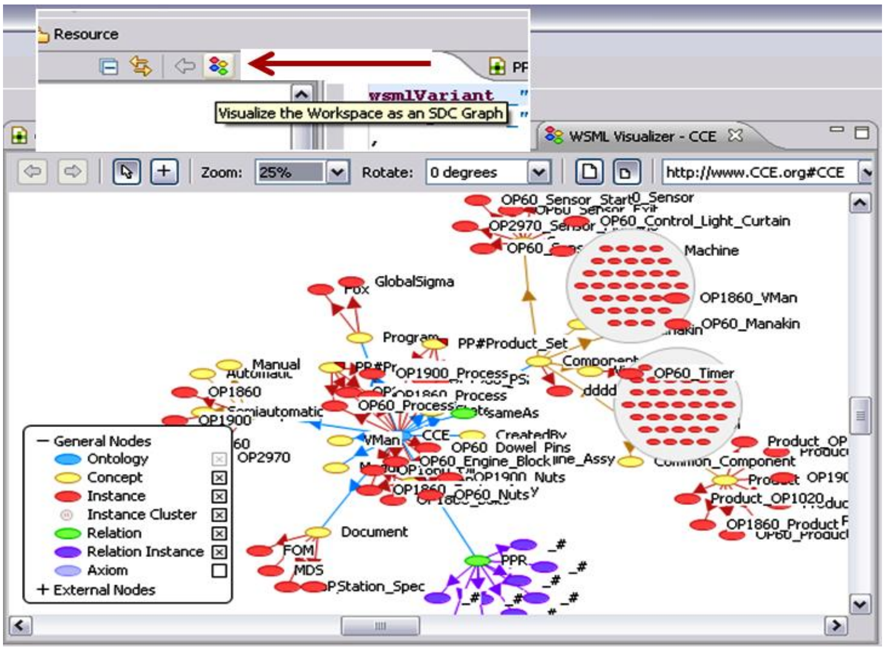
Figure 97: The WSMT editing area

This area in the IDE environment serves as the main knowledge capturing and ontology generating place.

WSML Visualiser

The ontology generated in the WSMT editor can also be seen as a SDC[97] graph which is automatically generated on the basis of asserted model of the ontology as shown in Figure 98:

[97] Semantic Discovery Caching



Zoom: 25%

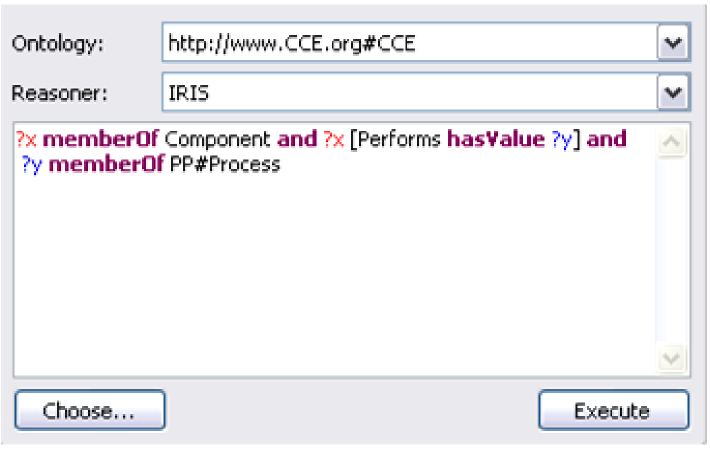
Rotate: 0 degrees

http://www.CCE.org#CCE

Figure 98: A screenshot of SDC graph in WSMT

WSMT Reasoner

Just below the WSML Navigator window, the user can access the WSML reasoner. The reasoning capabilities in the ontology are provided through a dedicated reasoner called IRIS reasoner, a snapshot is shown in Figure 99.



Ontology: http://www.CCE.org#CCE

Reasoner: IRIS

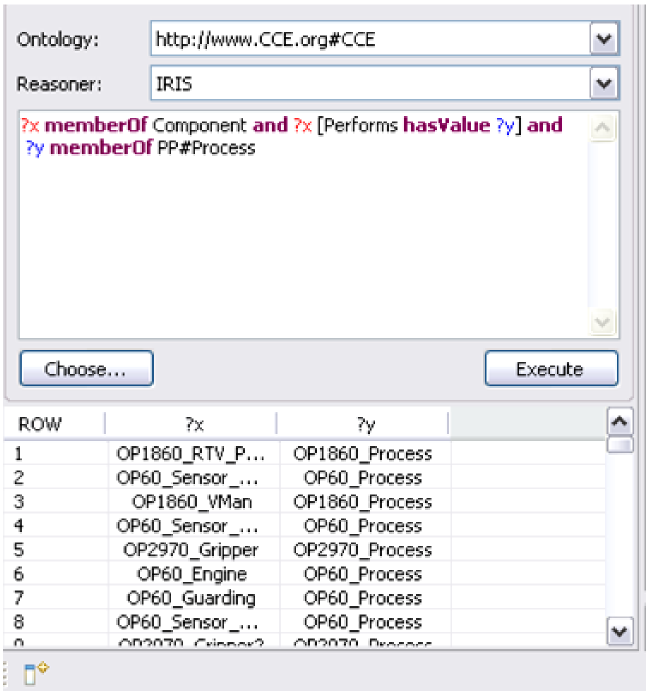
```
?x memberOf Component and ?x [Performs hasValue ?y] and
?y memberOf PP#Process
```

Figure 99: The WSMT reasoner

Integrated Rule Inference System (IRIS) serves as an extensible reasoning framework for WSML-Core and WSML-Flight variants of the WSML[98]. The reasoner helps to deduce new knowledge from already existing knowledge and provide logical reasoning in the developed system. The reasoner also testifies the consistency in the ontology and serves to provide conclusions for the user defined axioms in the ontology. The reasoner uses logic programming based query format to manipulate input information.

Direct Query – Search

The querying in the WSMT editor is logic based (Rule based) query through reasoner i.e. IRIS reasoner. Though WSML is an extension of RDF in many ways yet it is not an RDF based query such as SPARQL. The query in WSML is based on 1st order logic / logic programming.



Ontology: http://www.CCE.org#CCE

Reasoner: IRIS

```
?x memberOf Component and ?x [Performs hasValue ?y] and
 ?y memberOf PP#Process
```

ROW	?x	?y
1	OP1860_RTV_P...	OP1860_Process
2	OP60_Sensor_...	OP60_Process
3	OP1860_VMan	OP1860_Process
4	OP60_Sensor_...	OP60_Process
5	OP2970_Gripper	OP2970_Process
6	OP60_Engine	OP60_Process
7	OP60_Guarding	OP60_Process
8	OP60_Sensor_...	OP60_Process
9	OP2970_Gripper2	OP2970_Process

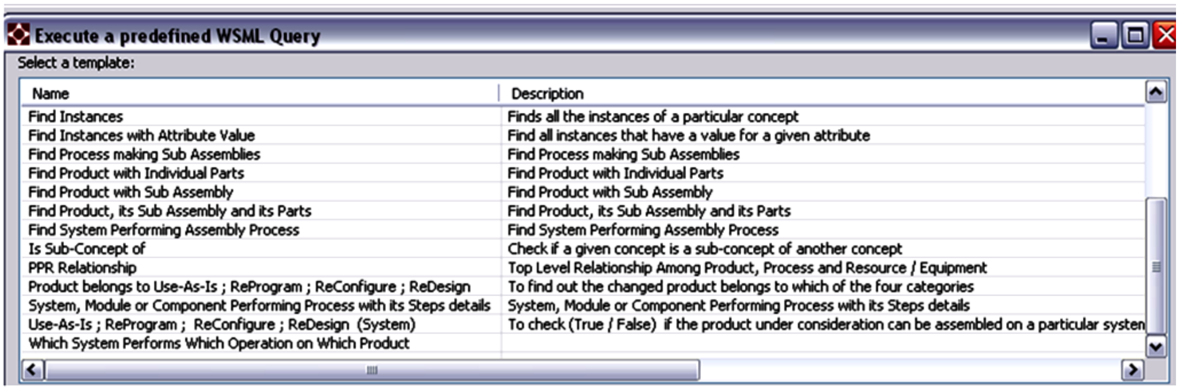
Figure 100: Querying ontology directly in WSMT

The Figure 100 shows one of the results of the query with results, the query is based on Logic Programming (LP) syntax.

[98] IRIS – supporting reasoning over WSML ontologies An open source project: http://sourceforge.net/projects/iris-reasoner/

Query Template

In addition to the direct input to the reasoner, a user friendly template is provided to help users to find the relevant type of query in the correct LP format. The template resides all types of queries which a user may require along the course of assembly line design knowledge requirement so that the users may select the required search from already available queries without knowing the actual syntax. The selected query will be transferred into the reasoner input area. Then the user may change only the main concepts as per requirement, keeping the syntax of the query unchanged so as to use the template as a generic guideline for all types of possible queries. The query template is shown below in Figure 101.



Select a template:

Name	Description
Find Instances	Finds all the instances of a particular concept
Find Instances with Attribute Value	Find all instances that have a value for a given attribute
Find Process making Sub Assemblies	Find Process making Sub Assemblies
Find Product with Individual Parts	Find Product with Individual Parts
Find Product with Sub Assembly	Find Product with Sub Assembly
Find Product, its Sub Assembly and its Parts	Find Product, its Sub Assembly and its Parts
Find System Performing Assembly Process	Find System Performing Assembly Process
Is Sub-Concept of	Check if a given concept is a sub-concept of another concept
PPR Relationship	Top Level Relationship Among Product, Process and Resource / Equipment
Product belongs to Use-As-Is ; ReProgram ; ReConfigure ; ReDesign	To find out the changed product belongs to which of the four categories
System, Module or Component Performing Process with its Steps details	System, Module or Component Performing Process with its Steps details
Use-As-Is ; ReProgram ; ReConfigure ; ReDesign (System)	To check (True / False) if the product under consideration can be assembled on a particular systen
Which System Performs Which Operation on Which Product	

Figure 101: A user friendly query template

Before these hierarchical classifications and taxonomical arrangements, it was difficult for the engineers especially the novice engineers to fully understand the atomic functionalities of decomposed machine tools.

Fast View / Outline Window

As the ontology is populated and concepts, properties, relations, axioms and instances are added, the editing area, obviously, becomes text heavy. To navigate through the generated ontology another window is provided on the right side of the editing area. This is called fast view / outline window which is used for fast navigation along the developed ontology as shown in Figure 102 below:

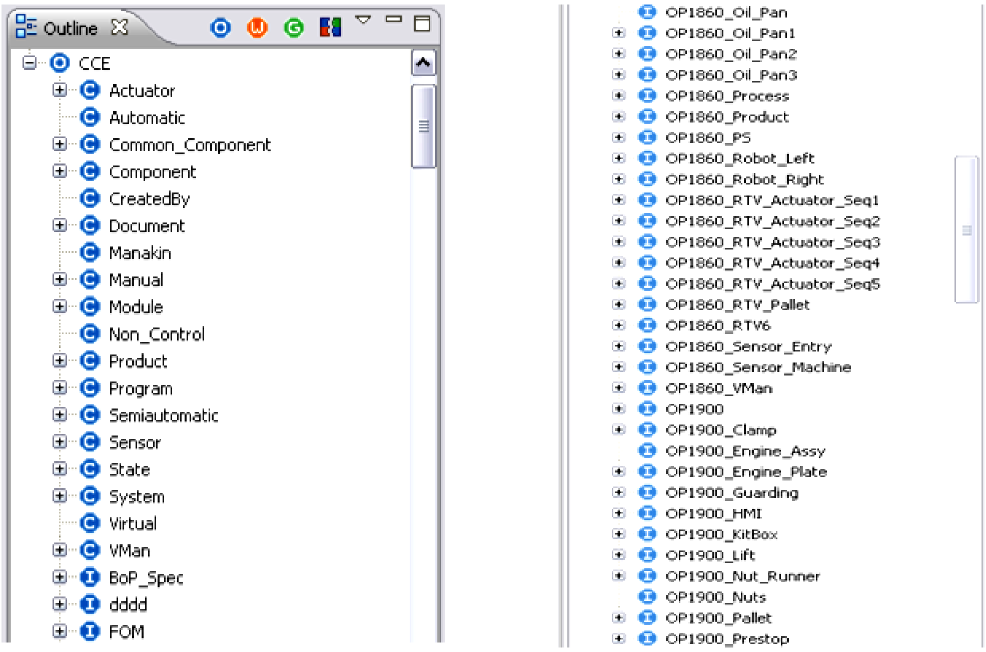
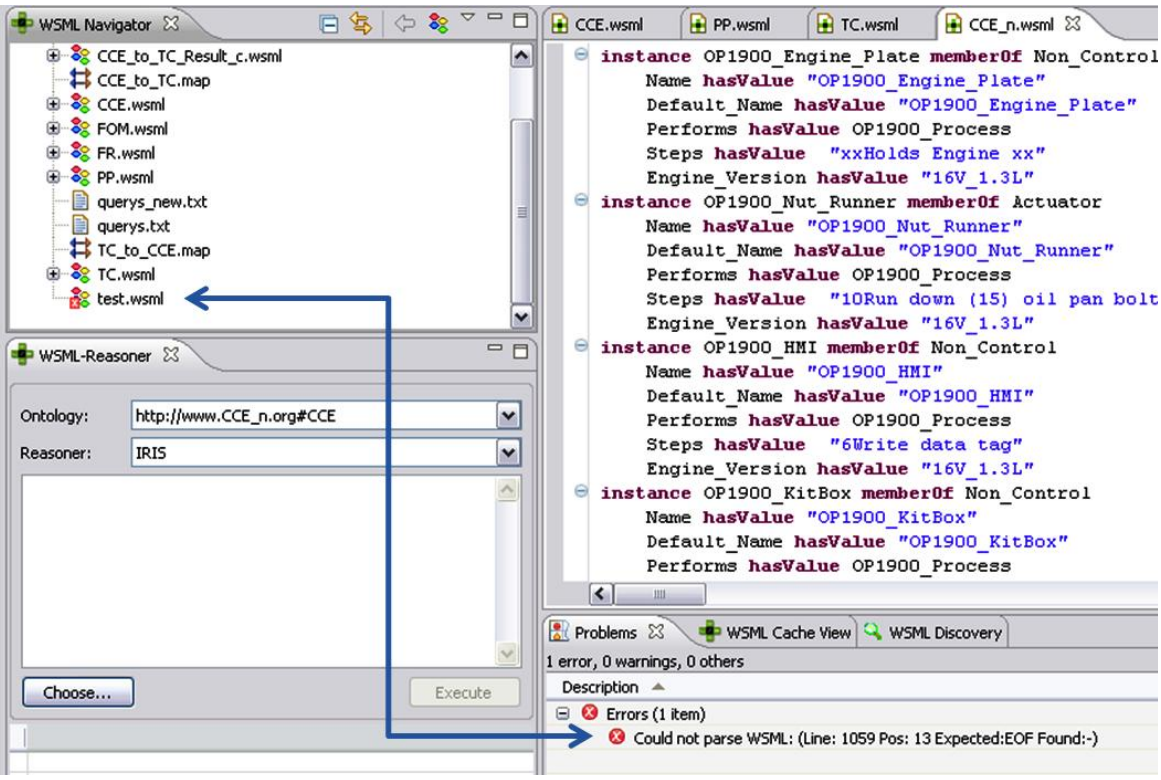


Figure 102: The fast view option in WSMT

The fast view window groups all the relevant information regarding a specific concept therefore it becomes easy to navigate through the developed ontology as described in Figure 102.

Consistency Checking

One of the benefits of the ontology is automatic consistency checking. The concepts, relations and properties defined at the abstract level must fulfil the consistency down to the instance level for each and every instance. The same facility is provided in WSMT i.e. warnings and errors are conveyed to the ontology developer. Any discrepancy and/or clerical mistake is not acceptable and is highlighted automatically with the exact line and text of the potential mistake in the 'Problems' window area right at the bottom of the editing area as shown in Figure 103.



Ontology: http://www.CCE_n.org#CCE

Reasoner: IRIS

```
instance OP1900_Engine_Plate memberOf Non_Control
    Name hasValue "OP1900_Engine_Plate"
    Default_Name hasValue "OP1900_Engine_Plate"
    Performs hasValue OP1900_Process
    Steps hasValue  "xxHolds Engine xx"
    Engine_Version hasValue "16V_1.3L"
instance OP1900_Nut_Runner memberOf Actuator
    Name hasValue "OP1900_Nut_Runner"
    Default_Name hasValue "OP1900_Nut_Runner"
    Performs hasValue OP1900_Process
    Steps hasValue  "10Run down (15) oil pan bolt
    Engine_Version hasValue "16V_1.3L"
instance OP1900_HMI memberOf Non_Control
    Name hasValue "OP1900_HMI"
    Default_Name hasValue "OP1900_HMI"
    Performs hasValue OP1900_Process
    Steps hasValue  "6Write data tag"
    Engine_Version hasValue "16V_1.3L"
instance OP1900_KitBox memberOf Non_Control
    Name hasValue "OP1900_KitBox"
    Default_Name hasValue "OP1900_KitBox"
    Performs hasValue OP1900_Process
```

```
1 error, 0 warnings, 0 others
Errors (1 item)
Could not parse WSML: (Line: 1059 Pos: 13 Expected:EOF Found:-)
```

Figure 103: Errors and inconsistencies displayed in WSMT

In general, WSMT does not allow users to build incomplete, inconsistent and ambiguous knowledge elements. This is normally an advantage, nevertheless, there may be some extraordinary situation where inconsistency may be required to occur and acceptable as part of some special phenomenon inherent to the assembly line[99]. However, this is not allowed in ontological architecture due to reasoning error encountered when defining asserted model of the ontology. Any inconsistency needs to be resolved at the abstract level and individuals at the concrete level need to follow the abstract logic.

[99] A particular component of a machine occurring twice at the same station with the same name and the same function, which is not required to be distinguished categorically, (can be distinguished automatically at the process level and not at the component level). At abstract level, it may be declared that a component can perform more than one step, but the steps must be unique. The steps may not be unique if carried out at different times. To cater for this, another concept 'time' needs to be defined in the resource properties to cater for the temporal aspects. As this is an extremely rare case therefore it is been left for refinement at later stage. This is one of the limitations of the current system.

Appendix 2.

Ontology of the Festo Rig

Building a Generic Model of the Festo Rig

Ontology building process starts with defining the generic model of the domain. This defines the major elements their properties and relationships among them as well as basic hierarchy structure. Once the generic model is correctly defined, the next step is to define the specific model i.e. define the instances of the existing domain of interest.

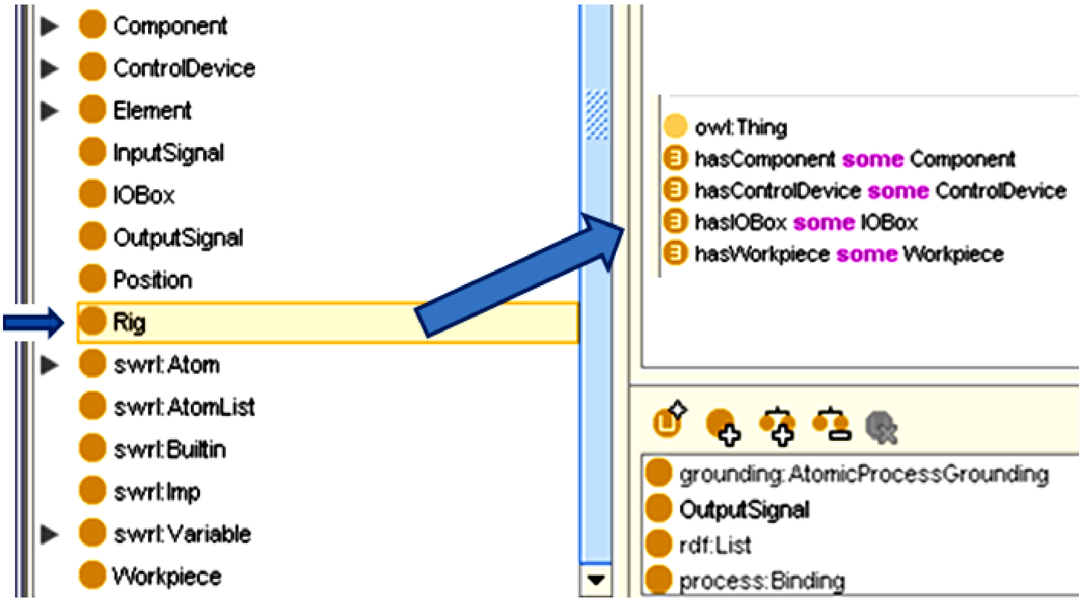
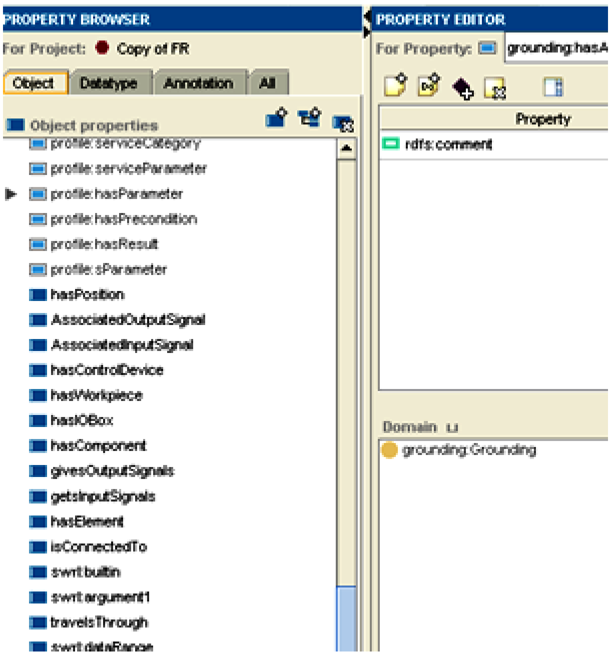


Figure 104: Definition of the class 'Rig' in Protégé ontology editor

The Figure 104 shows the definition of the concept 'Rig', the Rig consists of some components as building blocks, it has a control device, IO, and a workpiece. Similarly components of the Festo rig were defined with their properties and relations.

Defining properties of the classes: Once the rig is defined then the properties of the components are added as well as relations among different components established as shown in Figure 105.



PROPERTY BROWSER

For Project: Copy of FR

Object properties

profile:serviceCategory
profile:serviceParameter
profile:hasParameter
profile:hasPrecondition
profile:hasResult
profile:sParameter
hasPosition
AssociatedOutputSignal
AssociatedInputSignal
hasControlDevice
hasWorkpiece
hasIOBox
hasComponent
givesOutputSignals
getsInputSignals
hasElement
isConnectedTo
swrl:builtin
swrl:argument1
travelsThrough
swrl:dataRange

PROPERTY EDITOR

For Property: grounding:hasA

Property
rdfs:comment

Domain ⊔

grounding:Grounding

Figure 105: Definition of ‘Properties’ of defined classes

Creating ‘Instances’ of the defined Classes

Once the generic model of the domain is complete, the next step is to define the particular instances of the model of interest. Following the same principle, once the generic model of the Festo rig was constructed using abstract classes, instances of the classes were defined in protégé editor. This instance of the class depicts the actual rig with its characteristics and can be used to communicate with the other real world objects as shown in Figure 106.

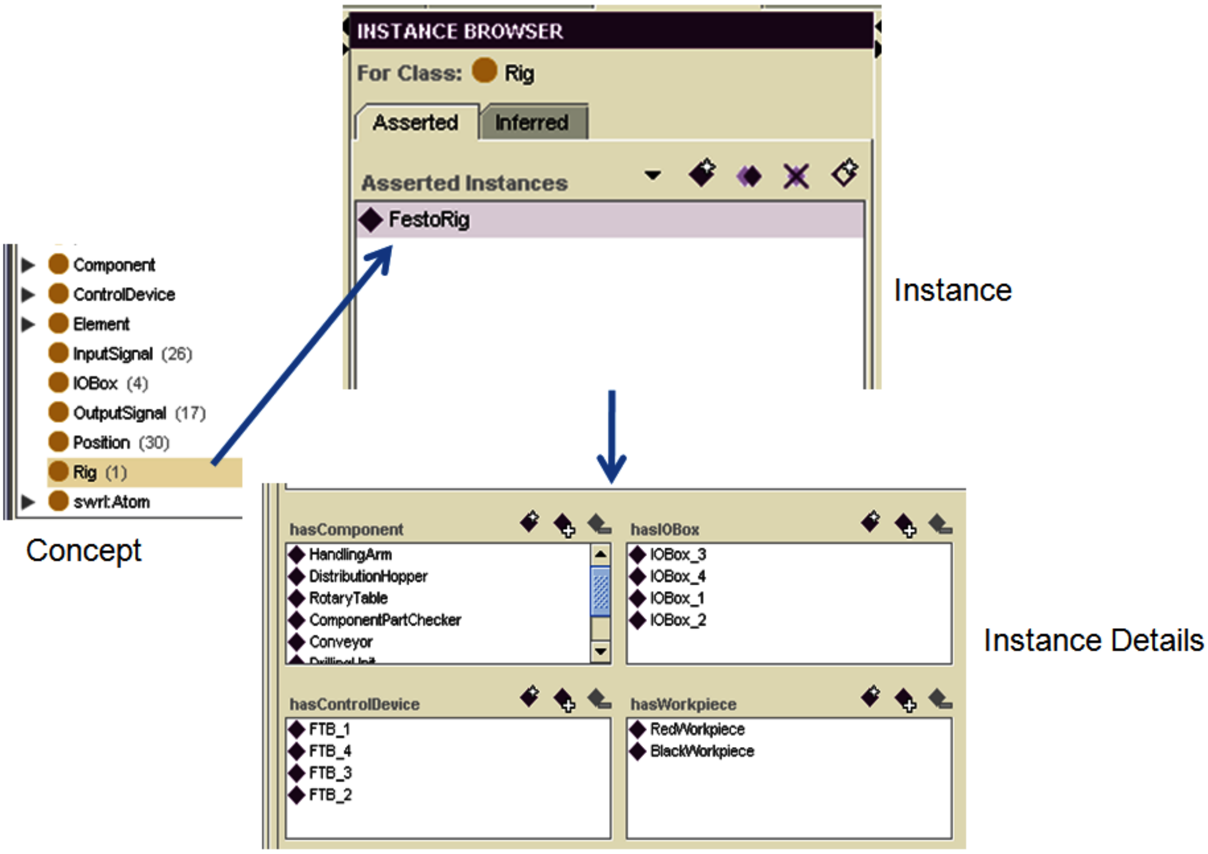


Figure 106: Creating instance of the defined class ‘Rig’ in Protégé ontology editor

Once all the instances have been created, asserted or inferred model of the Festo rig can be generated and analysed. This could be logically proved to be true with the help of reasoner provided the generic model has been created correctly. The Figure 107 shows the inferred model of the rig:

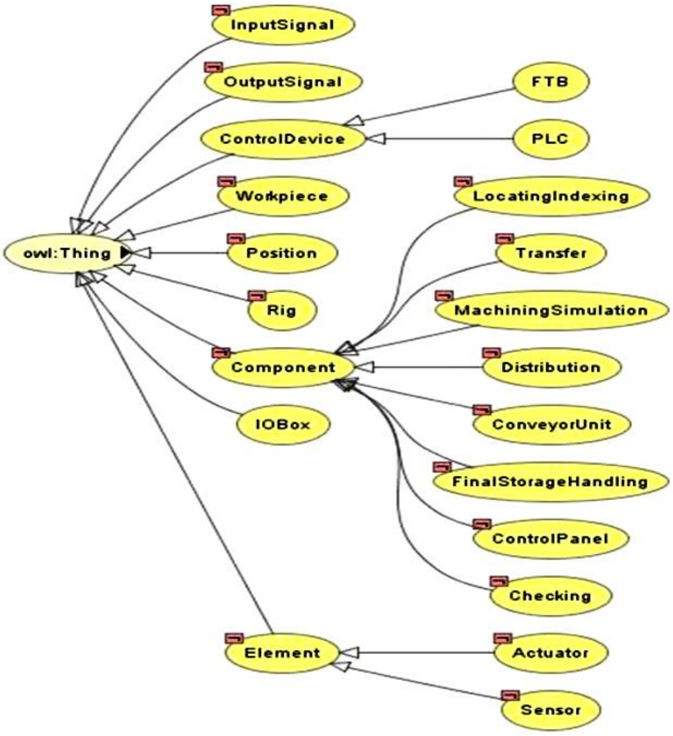


Figure 107: The hierarchical representation of the Festo rig – inferred model

One of the important elements in the ontology of the Festo rig was to create axioms and rules. For this two concepts were introduced i.e. 'workpiece characteristics' and 'workstation characteristics' and a few elementary key characteristics were defined as shown in Figure 108:

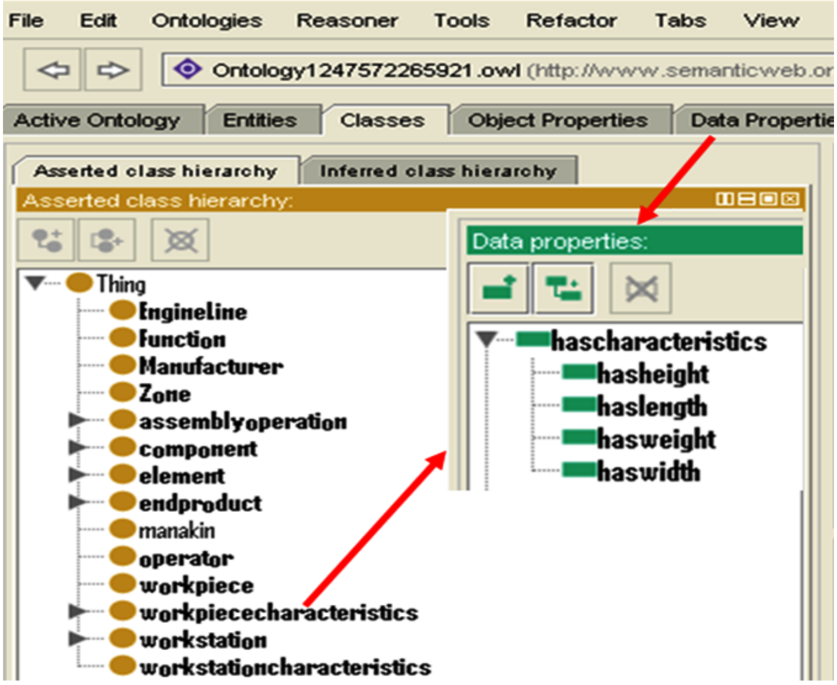


Figure 108: Relational constraints in Festo rig

Workpiece i.e. product length, width, height and weight is defined as data property. Similarly workstation is defined to have weight lifting capability as well as possible limitations in X, Y and Z axes.

Appendix 3.

Case Study – Engine Fit Analysis

The analysis of OP60 has been provided in section 6.6.3. For the author's case study, four stations were investigated, the stations OP1860, OP1900 and OP2970 are described here.

OP1860 System (RTV Station)

The next workstation investigated is the OP1860. Again this is a semi-automatic assembly station with a combination of two robotic arm units, an operator and a system carrying engine sub-assembly on the pallet passing by and stopping in front of the operator. Operation 1860 system is also called RTV station in which sealant is used as a gasket to seal oil pan sump with rest of the engine sub-assembly. At Ford's Dagenham Tiger Assembly Plant, this station consists of two robotic arms which are programmed to apply RTV sealant on the profile of the oil pan. This part (oil pan) is then placed on rest of the sub-assembly through guided dowel pins and transported to the next station.

RTV (Room Temperature Vulcanising) silicone sealants are being used in a wide range of industries including automotive. RTV sealant is a silicone sealant with aerospace and military applications in addition to a wide range of industrial applications[100]. Diesel Resistant RTV Silicone Sealant with Acetone cure type is used on the Ford's OP1860 station. The Process is that RTV is applied to the oil pan and then oil pan is assembled to the engine. A snapshot of the RTV station in CCE tool is provided in Figure 109.

[100] http://www.acc-silicones.com/applications

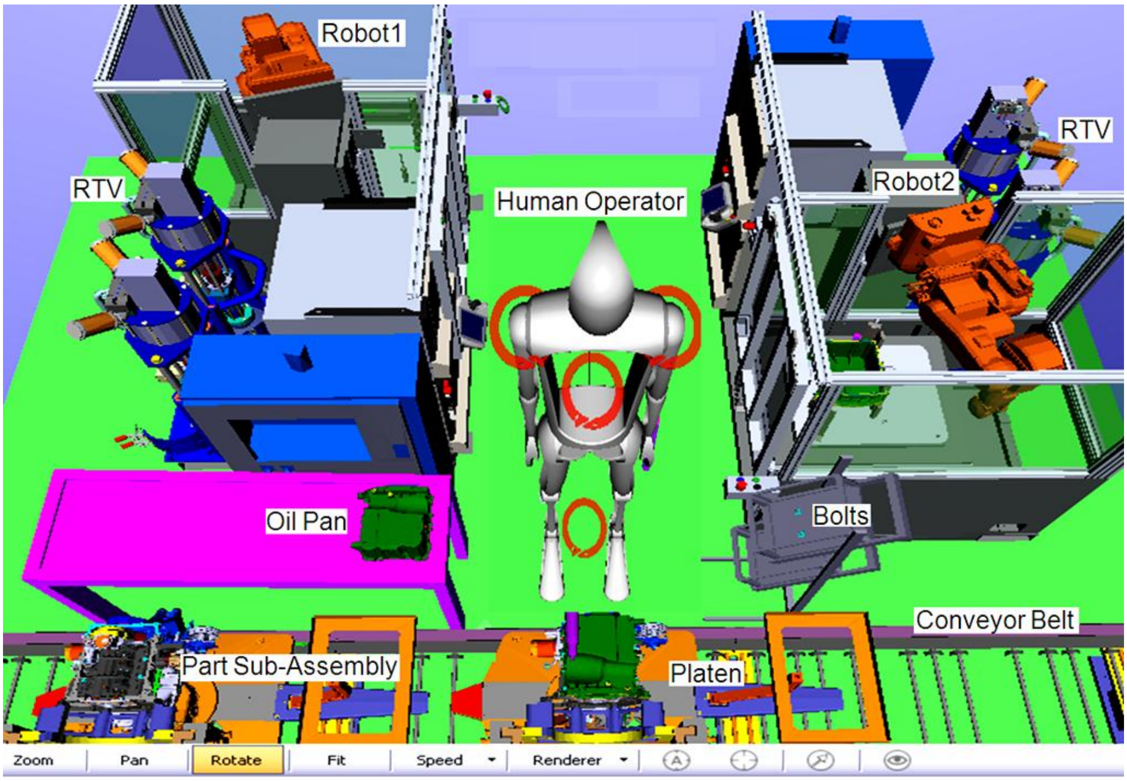


Figure 109: A snapshot of the RTV station modelled in the CCE tool

A brief of the assembly steps are shown below in Figure 110.

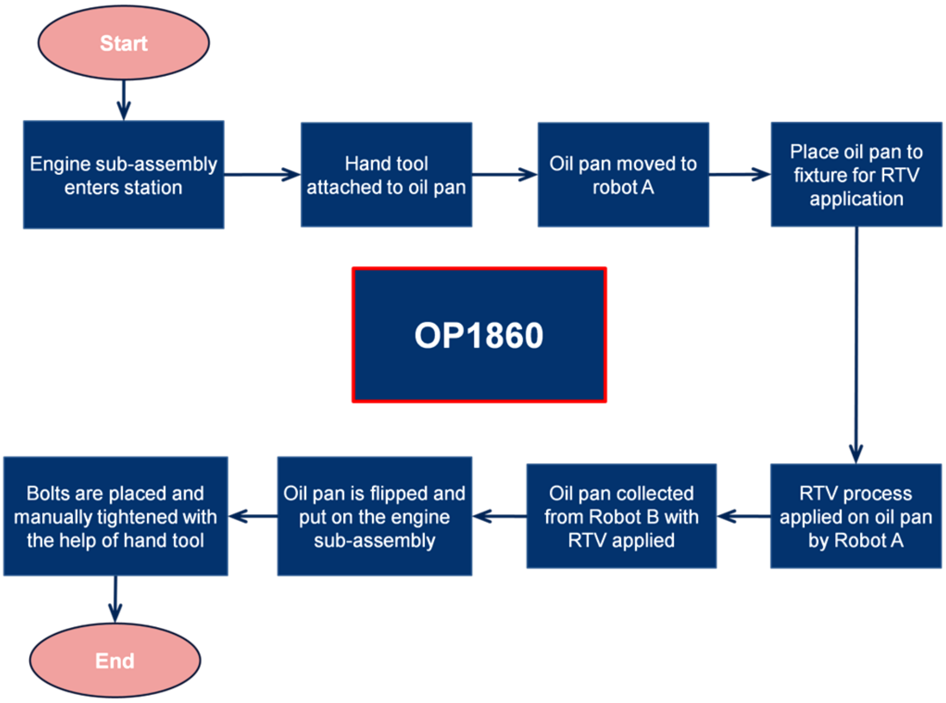

Start
Engine sub-assembly enters station
Hand tool attached to oil pan
Oil pan moved to robot A
Place oil pan to fixture for RTV application
OP1860
RTV process applied on oil pan by Robot A
Oil pan collected from Robot B with RTV applied
Oil pan is flipped and put on the engine sub-assembly
Bolts are placed and manually tightened with the help of hand tool
End

Figure 110: Assembly process sequence for OP1860

The operation consists of applying RTV sealant to seal oil pan sump with rest of the engine sub assembly. The robots are used to apply RTV on the complex contour of the mating surface on the oil pan sump. The operator makes sure that the oil pan sump is placed on the fixture correctly and transports the oil pan sump to and from robots and finally inserts nut runners into holes and applies initial torque with the help of the hand tool. The actual sequence is briefed in Figure 111.

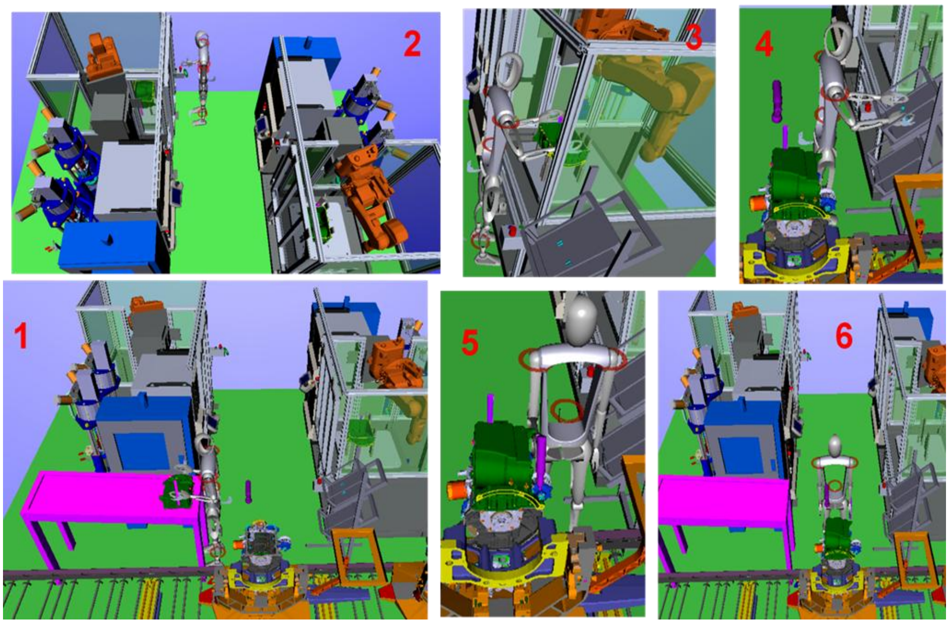


Figure 111: Simulation model of assembly station OP1860 generated by CCE tool

The operator picks up an oil pan and places it on the fixture of the robot1 for RTV application and picks up oil pan from the other robot2 which has already applied RTV and placed it to the part sub assembly on the platen. The operator then inserts bolts into the oil pan and tightens them with the help of a manual nut runner (the nuts will be tightened to the required final torque on the next station with automatic nut runner).

The engine sub-assembly enters into the system and stops at the specified location for assembly processes to begin. The operator picks up the oil pan sump, takes it to the Robot A and places on the fixture of the Robot A as depicted in part ‘1 & 2’ of the Figure 111. On the way back, the operator would collect the RTV applied oil pan sump from Robot B as shown in part ‘3’ of Figure 111, and place it on the engine sub-assembly. (The next oil pan will be applied RTV sealant on Robot B and the sealant applied sump will be collected from Robot A). Once the sump is placed on the engine sub-assembly, the operator picks up bolts, as shown in part ‘4’ of Figure 111, inserts bolts into the sump to the engine sub-assembly and applies initial torque manually through nut runner hand tool, (semi-automatic torque wrench), as shown in part ‘5’ of Figure 111. As the operator completes the manual torque application, the assembly operation is complete on this system and the pallet leaves the station through conveyor to the next station as demonstrated in part '6' of Figure 111. It is a 57 second cycle time station.

Again for this workstation, KCs are defined which control the assemblability of the parts on this station, the KCs are derived from the following parts on the workstation as shown in Figure 112.

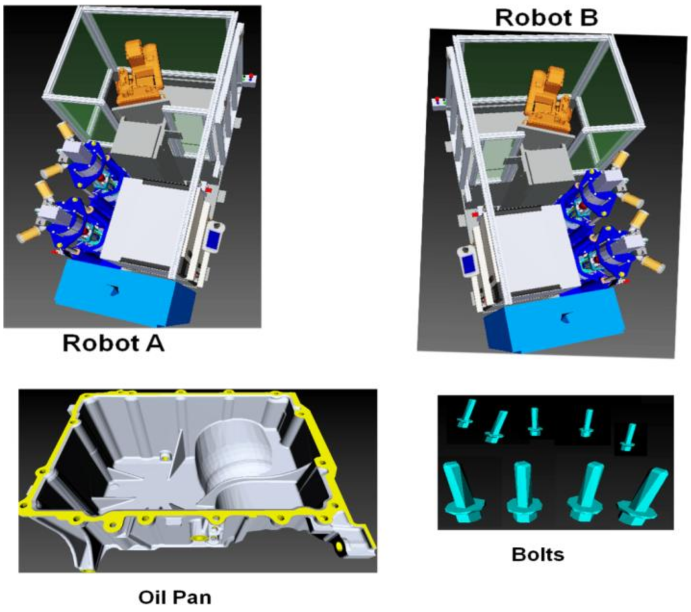


Figure 112: OP1860 system with product parts and components

Requirement: Retrieve parts (product_set) associated with oil pan i.e. which parts make sub-assemblies with oil pan, then retrieve associated stations/systems and associated processes.

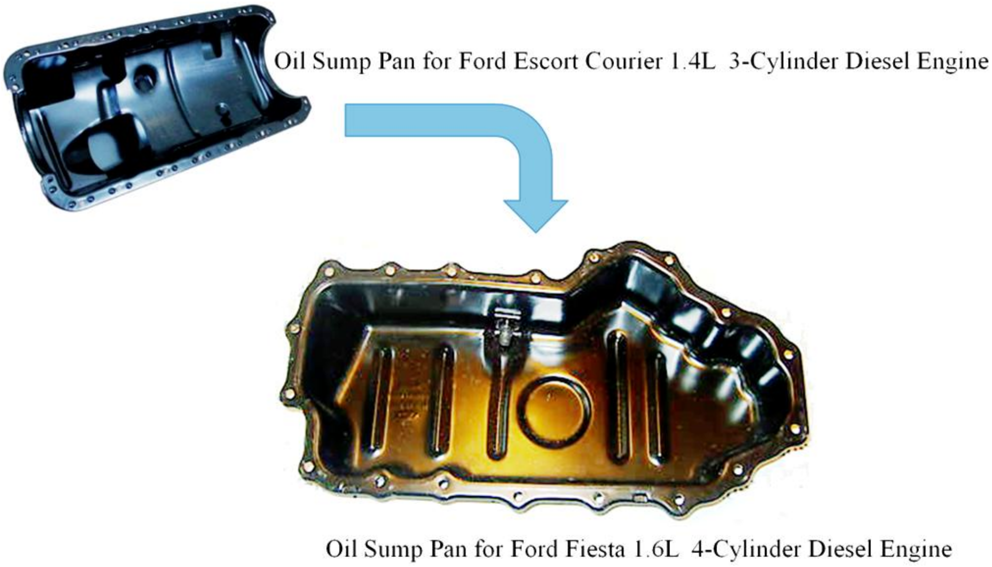

Oil Sump Pan for Ford Escort Courier 1.4L 3-Cylinder Diesel Engine
Oil Sump Pan for Ford Fiesta 1.6L 4-Cylinder Diesel Engine

Figure 113: Change in cylinders results oil pan sump change from 1.4L to 1.8L diesel engine

Key Characteristics (KCs) for the RTV Station i.e. OP1860

There are nine product key characteristics defined at the OP1860 station as shown in Table 34. As the engine size increases, the engine oil volumetric capacity must increase i.e. size of oil pan sump needs to be increased.

KC	ofType	Property	Existing Value	New Value
OP1860_Bolts_Qty	Integer	hasValue	10	10
OP1860_Diameter_of_Bolts	Integer	hasValue	06	06
OP1860_Length_of_Bolts	Integer	hasValue	20	20
OP1860_Oil_Pan_ Length	Integer	hasValue	325	435
OP1860_Oil_Pan_Width	Integer	hasValue	310	305
OP1860_Oil_Pan_Depth	Integer	hasValue	140	160
OP1860_Product_ Length	Integer	hasValue	500	610
OP1860_Product_Width	Integer	hasValue	450	445
OP1860_Product_Height	Integer	hasValue	600	590
Engine_Version	String	hasValue	1.4L Diesel	1.8L Diesel
OP1860_RTV_Profile			x	x’

Table 34: KCs of OP1860 System

As described in the chapter 6, section 6.7.4, one of the limitations of the current KB system is its inability to transform contour profile into measureable form as shown in Figure 114.

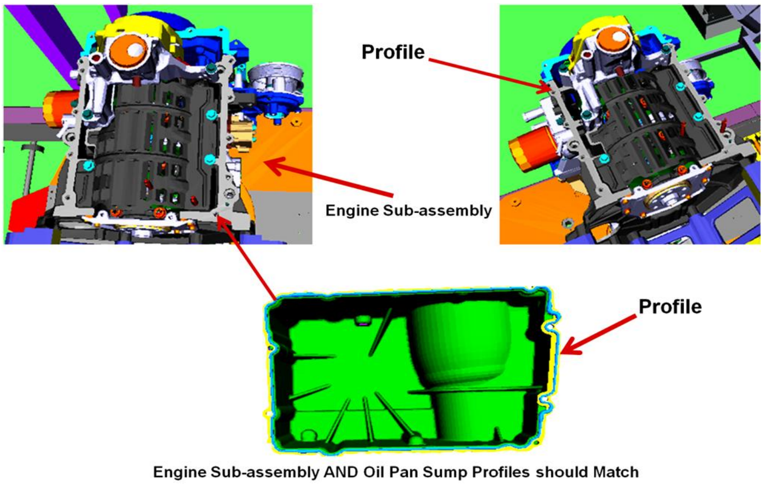


Figure 114: The profiles of the engine sub-assembly and oil pan sump should match

The KCs with their attributes along with rules are described in Table 35.

KCs	Existing attribute	New / Changed attribute	Rules / Constraints	
			For Robot	For System
OP1860_Bolt Qty Diameter Length	 10 6 20	 10 6 20	If qty=10 and dia=6 and length=20 THEN Same Bolts ELSE IF dia≠6 OR length≠20 THEN Redesign Bolts	

OP1860_Oil_Pan OP1860_Oil_Pan_Length OP1860_Oil_Pan_Width OP1860_Oil_Pan_Depth	325 310 140	435 305 160*	If L=325 then use as is Else Re-program	If L<500 then use as is Else If L > 500 and then Redesign
			W=310 then use as is Else Re-program	If W<450 then use as is Else If 450<W<1000 Then Reconfigure Else If W > 1000 then Redesign
			D=140 then use as is Else Re-program	If D<500 then use as is Else If 500<D<1000 Then Reconfigure Else Redesign
Profile of oil pan AND Profile of mating sub-assembly			As width changes therefore profile is changed	

Product length x width x height
(clearance in x, y, z) With oil pan
	Length = 500
	Width = 450
	Height = 600
		Length = 610
		Width = 445
		Height = 590
		Height of engine sub-assembly is less however depth of oil pan is greater
			IF length ≤ 650 AND width ≤ 450 AND height ≤600 THEN Use-as-is
			ELSE IF (?length > 650 and ?length <750) AND (?width > 450 and ?width < 655) or (?height>600 and ?height< 765) THEN Reconfigure
			ELSE IF (?length > 750) or (?width > 655) or (?height>765) THEN Redesign**	

Table 35: Rules and axioms formulated on the KCs of OP1860 System

*Due to increase in cylinders, the engine oil needs to be increased therefore, depth of oil pan sump is slightly increased.

**Most of the values are obtained from simulation results however a few are obtained from Ford engineers' tacit knowledge

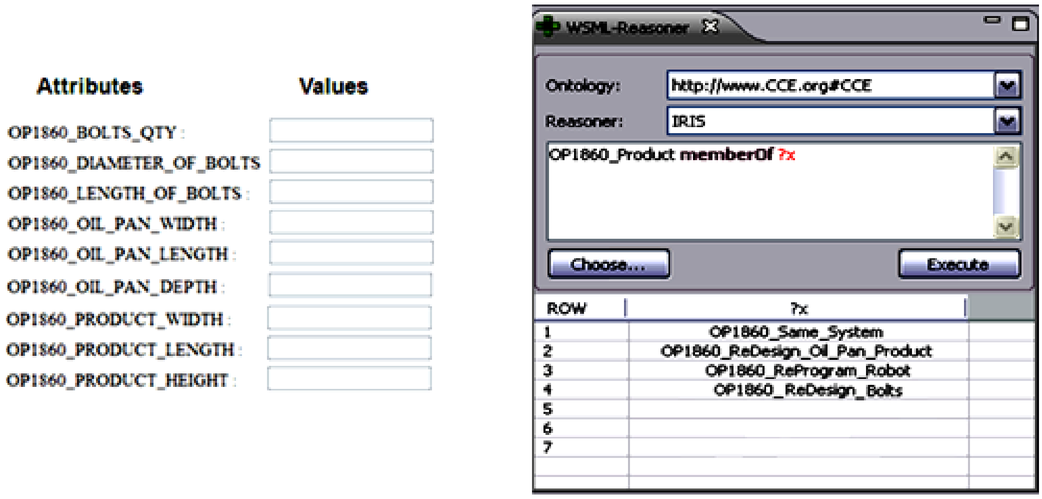
The contour of the profile cannot be translated into numerical numbers which is one of the current limitations. Once the rules are formulated, the next step is to provide input values for the new / changed product in terms of key characteristics of products and resources. The KB system has got the changed parameters of the product for the 'oil pan run down' product build stage (product-set) to be compared with the existing parameters for the system OP1860. These two KCs can be compared through defined rules and axioms. When the query is run, the system has got both the set of parameters (existing vs. new) and it will compare the two values for each KC defined and provide the output, e.g. if the user wants to know the effects of the changed product on the existing system, the results may be obtained as shown below in Table 36.

OP1860 (RTV Station)		
Input		**Output**
OP1860_Bolt_Qty	10	OP1860 Same System
OP1860_Bolt_Diameter	6	
OP1860_Bolt_Length	20	OP1860 Redesign Product
OP1860_Oil_Pan_Width	435	
OP1860_Oil_Pan_Length	305	OP1860 Reprogram robot
OP1860_Oil_Pan_Depth	160	
OP1860_Product_Length	610	

OP1860_Product_Width	445	OP1860 Redesign Bolts
OP1860_Product_Height	590	

Table 36: Summary of Inputs and outputs for OP1860 in ALDIMS

The new / changed values of the KCs can be provided through web based form or by direct input into the WSMT editor, a snapshot of input parameters and output from the ALDIMS is shown below in Figure 115.



Attributes	Values
OP1860_BOLTS_QTY :	
OP1860_DIAMETER_OF_BOLTS	
OP1860_LENGTH_OF_BOLTS :	
OP1860_OIL_PAN_WIDTH :	
OP1860_OIL_PAN_LENGTH :	
OP1860_OIL_PAN_DEPTH :	
OP1860_PRODUCT_WIDTH :	
OP1860_PRODUCT_LENGTH :	
OP1860_PRODUCT_HEIGHT :	

Ontology: http://www.CCE.org#CCE

Reasoner: IRIS

```
OP1860_Product memberOf ?x
```

ROW	?x
1	OP1860_Same_System
2	OP1860_ReDesign_Oil_Pan_Product
3	OP1860_ReProgram_Robot
4	OP1860_ReDesign_Bolts
5	
6	
7	

Figure 115: Input KCs and the query output result by the KB system

As shown in the query output in Figure 108, the system suggests redesigning the product, reprogramming the robot and redesigning the bolts. As the depth of oil pan sump is increased, therefore bolts with larger length are required, similarly the robot needs to be reprogrammed as the depth of sump is increased, it needs to be adjusted to be placed on the fixture for applying RTV sealant.

OP1900 System

Similar to the previous two systems, in case of ‘OP1900 System’, the station consists of several functionally independent components. Key characteristics defined on this workstation are the quantity, diameter, length and torque of nut runner unit and product height, length, width etc. The product stage at this station is assembled product from the previous station and the addition of product parts i.e. nuts. A brief of the major steps defined on the station are shown in Figure 116.

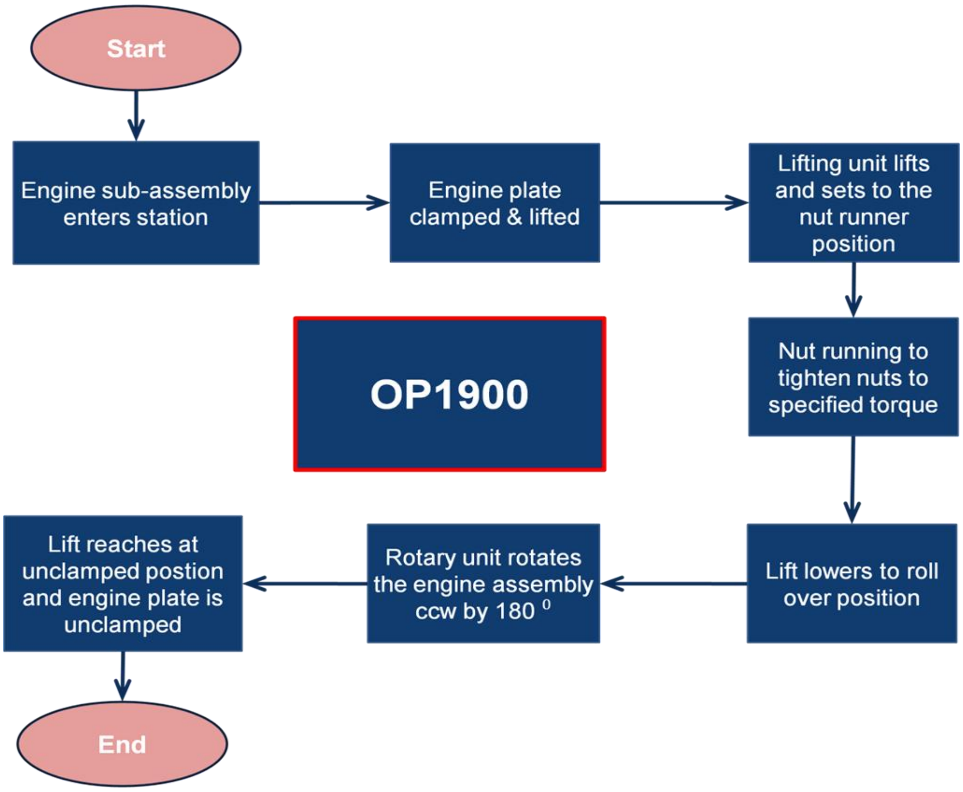

Start
Engine sub-assembly enters station
Engine plate clamped & lifted
Lifting unit lifts and sets to the nut runner position
OP1900
Nut running to tighten nuts to specified torque
Lift lowers to roll over position
Rotary unit rotates the engine assembly ccw by 180 0
Lift reaches at unclamped postion and engine plate is unclamped
End

Figure 116: OP1900 Assembly Process Sequence Steps

The system is decomposed into smaller independent functional units with the help of modular CCE tool, a snapshot of the important components of the system is shown below in Figure 117.

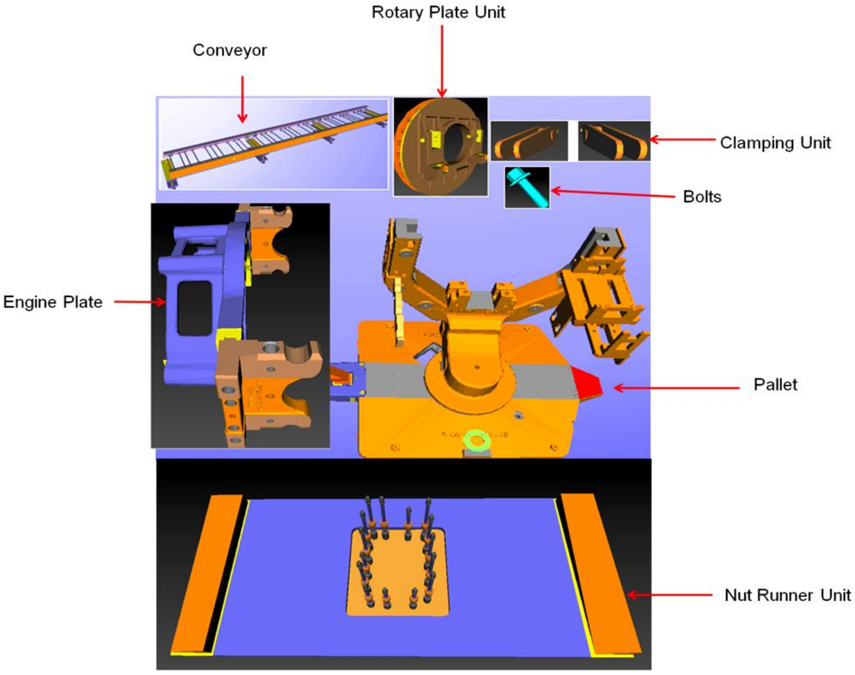


Figure 117: OP1900 system with components on the system

A few important components of the station consist of nut runner unit, clamping unit, engine plate, rotary plate unit etc. The nut runner unit is used to apply final torque to engine sub-assembly and oil pan sump. The clamping unit clamps the engine plate to the rotary plate unit.

OP1900 station is called ‘Oil Pan Rundown’ station. This is a fully automatic station and the major assembly operation performed on this station is the application of final torque to the nut runners already inserted into the proper place and manually tightened at the previous station. After tightening the nuts, the engine assembly is turned upside down. A few of the assembly process steps are shown below in Figure 118:

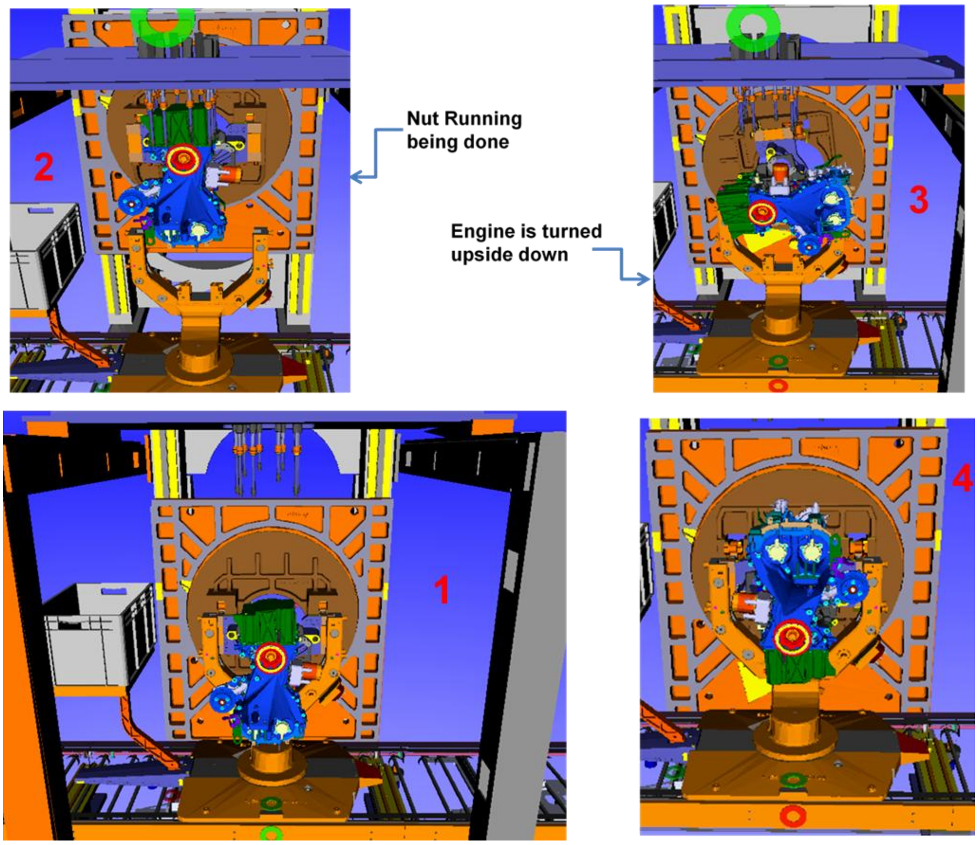


Figure 118: Simulation model of assembly station OP1900 generated by CCE tool

The engine sub-assembly enters into the system and stops at the specified location for assembly processes to begin. The ‘engine plate’, holding the engine sub-assembly and placed on the pallet, is clamped to the rotary plate to be lifted which is shown in part ‘1’ of the Figure 118. The clamped engine plate is lifted to nut runner working position so that nut running can be carried out as shown in part ‘2’ of the Figure 118. The engine plate is lowered to a certain height once nut running is complete and is rotated upside down ccw by 180^{0} as shown in part ‘3’ of the Figure 118. After the engine sub-assembly is rotated, the clamped engine plate is further lowered to be placed on the pallet as shown in part ‘4’ of the Figure 118. The assembly operation is complete on the system and the pallet leaves the station through conveyor to the next station.

As the four-cylinder engine width and height both are varied therefore, nut runners need further investigation. The height of the four-cylinder engine is reduced, however, the depth of the oil pan sump is also increased giving an overall net decrease in height to be 10mm as shown in Figure 119.

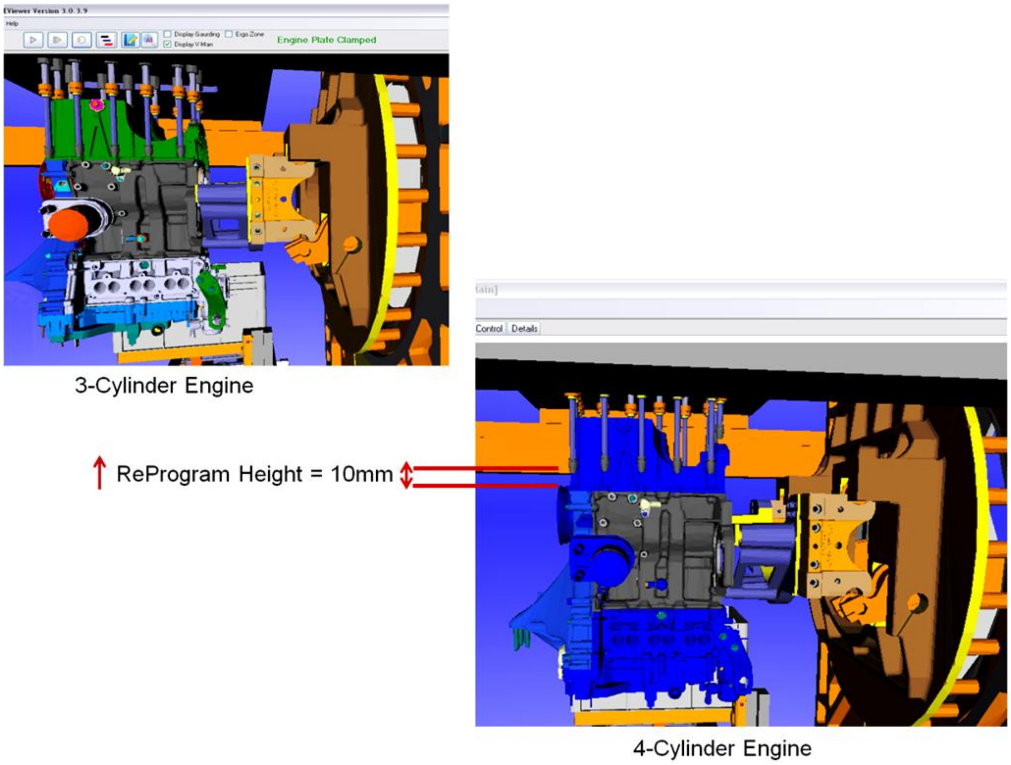


Figure 119: Nut runner height to be reprogrammed

The configuration of the existing nut runners is shown in Figure 119. As the engine width is increased, the position of nut runners needs to be re-adjusted. The length / height cannot increase more than 700mm as the engine assembly is rotated upside down, therefore, if length / height increases more than 700, the 180° rotation will not be possible unless redesigned. As the quantity and diameter of bolts is same, the only change is in length which is increased, therefore, the system has suggested reprogramming the nut runners. However, the position of the bolts is also changed due to change in dimensions of the engine, therefore the rule needs amendment. The dimensions of the engine directly affect the nut runners. The Figure 120 represents the current configuration of the nut runners.

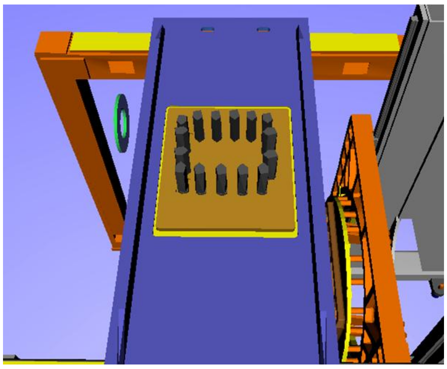


Figure 120: Configuration of the existing nut runners

It implies that if engine height changes then reprogram the nut runners and if engine width / length changes then reconfigure the nut runners.

The KCs of engine sub-assembly and the system with their attributes, rules and axioms are tabulated below in Table 37:

KCs	Existing attribute	New / Changed attribute	Rules / Constraints
OP1900_Clamp	No effect		
OP1900_Nut_Runner Qty Diameter Length 1 Length 2 Torque Configuration	15 6 20 75 10	15 6 20 65 10	If ?Qty=15 and ?Diameter=6 ?Length1=20 and ?Length 2=75, and ?Torque=8 – 11 THEN Same NutRunner ELSE IF (?qty = 15 and ((?dia > 6 and ?dia =< 25) or ?length1 !=20 or ?length2!=75 or torque <11)) THEN Reprogram NutRunner ELSE IF (?qty != 15 and ((?dia > 25) or ?length1>75 or ?length2>125 or torque >11)) THEN Reconfigure NutRunner ELSE IF (?qty != 15 and (?dia > 25 or ?length1>75 or ?length2> 125 or torque >11)) THEN Redesign NutRunner
The engine is being extended away from the station i.e. towards the operator, therefore the clearance between engine and guarding = 250mm minimum in any plane	If engine width is increased 250mm, it can accommodate without hindering guarding and greater than 250mm with slight modification in the guarding area.		IF ?width > 750 THEN Reconfigure the guarding

Product Length, Width Height	Length =500
	Width = 450
	Height = 600
		Length =610
		Width=445
		Height = 590
			IF length ≤650 AND width ≤600 AND height≤650 THEN Same System ELSE IF ((?length > 650 and ?length < 750) or (?width > 600 and ?width < 700) or (?height>650 and ?height< 700)) THEN Reprogram System ELSE IF ((?length ≥ 750) and (?width < 700 and (?height > 700 and height < 880)) THEN Reconfigure System ELSE IF (?length >750 or

			?width > 700 or ?height ≥ 880) THEN Redesign System

Product Width, Length and Height affects Nut Runners	Length =500
	Width = 450
	Height = 600
			IF Height = 600 AND (length =500 and width = 450) THEN Use-as-is nut runners
			IF Height ≠ 600 AND (length =500and width = 450) THEN Reprogram nut runners
			IF (length ≠ 500 or width ≠ 450) THEN Reconfigure nut runners (whatever height)

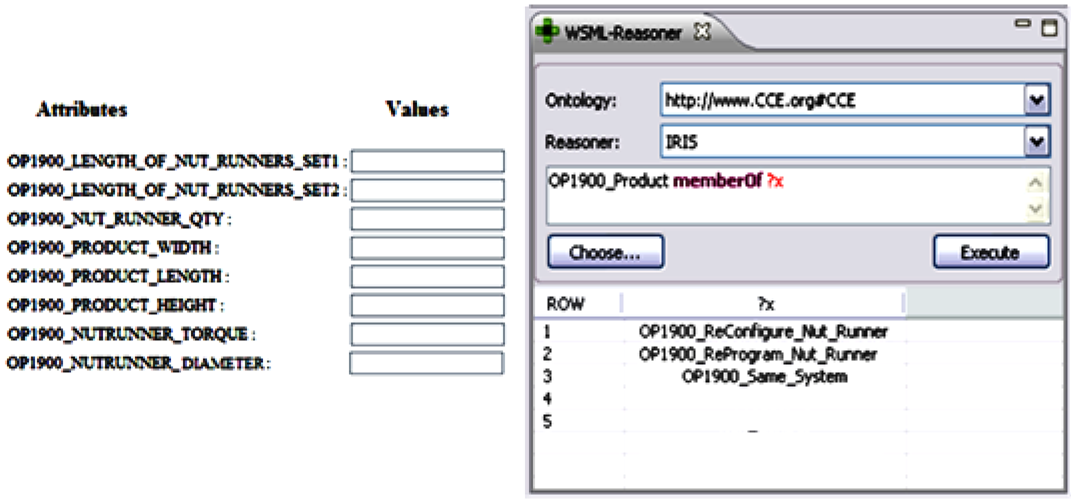
Table 37: Rules and axioms formulated on the KCs of OP1900 System

Once the rules are formulated, the next step is to provide input values for the new / changed product in terms of key characteristics of products and resources. The KB system has got the changed parameters of the product for the ‘oil pan run down’ product build stage (product-set) to be compared with the existing parameters for the system OP1900. These two KCs can be compared through defined rules and axioms. When the query is run, the system has got both the set of parameters (existing vs new) and it will compare the two values for each KC defined and provide the output, e.g. if the user wants to know the effects of the changed product on the existing system, the results may be obtained as shown below in Table 38:

OP1900 (RTV Station)		
Input		**Output**
OP1900_Nut_Runner_Qty	15	OP1900 Reconfigure Nut Runners
OP1900_Nut_Runner_Diameter	6	
OP1900_Nut_Runner_Length 1	20	
OP1900_Nut_Runner_Length 2	65	
OP1900_Nut_Runner_Torque	10	OP1900 Reprogram Nut Runners
OP1900_Product_Length	610	
OP1900_Product_Width	445	OP1900 Same System
OP1900_Product_Height	590	

Table 38: Summary of Inputs and outputs for OP1900 in ALDIMS

The new / changed values of the KCs can be provided through web based form or by direct input into the WSMT editor, a snapshot of input parameters and output from the ALDIMS tool is shown below in Figure 121.



Attributes	Values
OP1900_LENGTH_OF_NUT_RUNNERS_SET1 :	
OP1900_LENGTH_OF_NUT_RUNNERS_SET2 :	
OP1900_NUT_RUNNER_QTY :	
OP1900_PRODUCT_WIDTH :	
OP1900_PRODUCT_LENGTH :	
OP1900_PRODUCT_HEIGHT :	
OP1900_NUTRUNNER_TORQUE :	
OP1900_NUTRUNNER_DIAMETER :	

Ontology: http://www.CCE.org#CCE

Reasoner: IRIS

OP1900_Product memberOf ?x

ROW	?x
1	OP1900_ReConfigure_Nut_Runner
2	OP1900_ReProgram_Nut_Runner
3	OP1900_Same_System
4	
5	

Figure 121: Input KCs and the query output result by the KB system for OP1900

As shown in the Figure 114, the result of the query, the system suggests to reprogram as well as reconfigure the nut runners due to change in configuration as well as change in height of the bolts after change in oil pan sump while there is no assembly hard point for other components as well as overall product dimensions and system space, therefore no further changes are required in the system.

OP2970 System

The penultimate workstation on the discussed engine assembly line is the engine off-load station i.e. OP2970 which is a fully automatic station. The station transfers the assembled engine from the main rail to the off-line rail which is in parallel to the main conveyor rail. The engines from the off-line rail are packed and despatched for assembly with the main car body. The assembly operation consists of several steps and a few are shown in Figure 122.

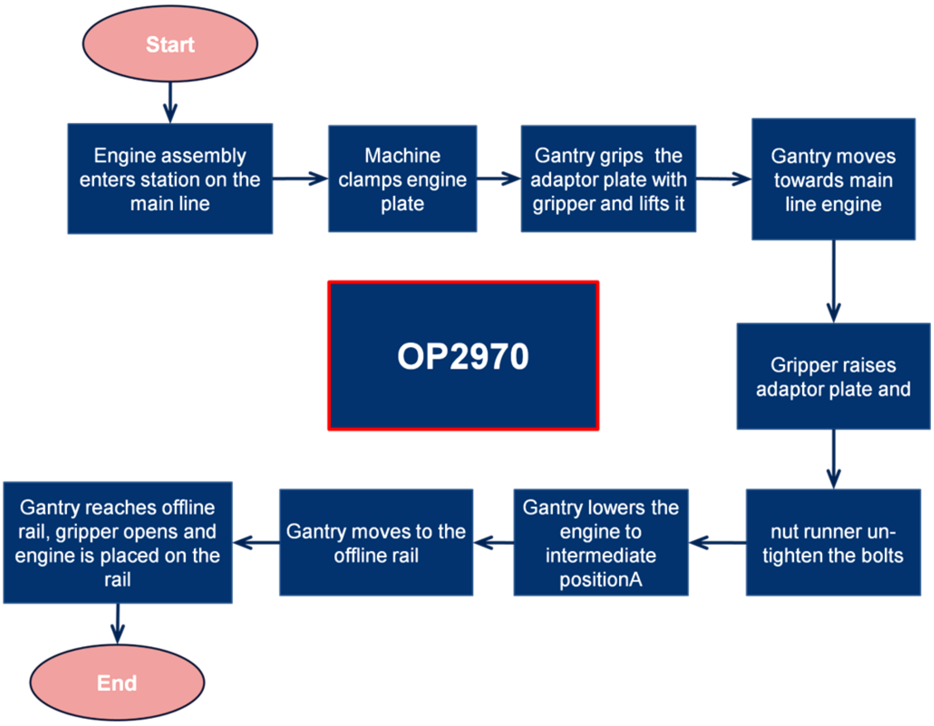

Start
Engine assembly enters station on the main line
Machine clamps engine plate
Gantry grips the adaptor plate with gripper and lifts it
Gantry moves towards main line engine
OP2970
Gripper raises adaptor plate and
nut runner un-tighten the bolts
Gantry lowers the engine to intermediate positionA
Gantry moves to the offline rail
Gantry reaches offline rail, gripper opens and engine is placed on the rail
End

Figure 122: OP2970 Assembly Process Sequence Steps

The engine enters into the workstation zone, the gantry grips and lifts the support plate and places underneath the engine assembly, the station unscrews the nut runners and the engine is placed on to the support plate which is then transported to the off-line conveyor with the help of the gantry as shown in Figure 123 below:

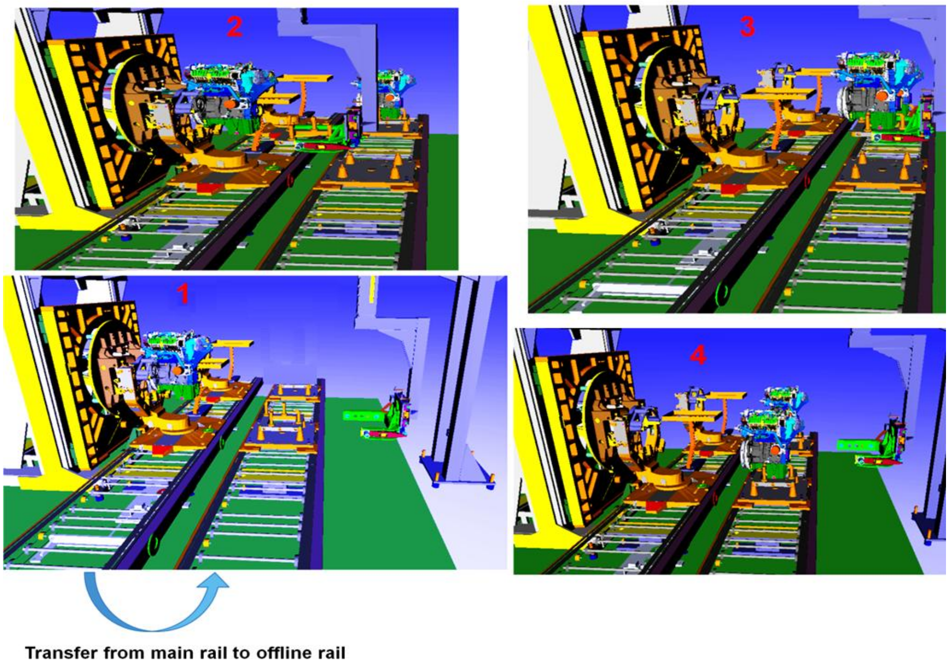


Figure 123: Simulation model of assembly station OP2970 generated by CCE tool

The engine sub-assembly enters into the system and stops at the specified location for assembly processes to begin. The ‘engine plate’, holding the engine sub-assembly and placed on the pallet, is adjusted in front of the nut runners so that the nut runners on this station may unclamped the final engine assembly from the engine plate as shown in part ‘1’ of the Figure 123. As the unclamping is under process, the gantry arm picks up the adaptor plate and moves to the main conveyor line and places the adaptor plate under the unclamping engine assembly as shown in part ‘2’ of the Figure 123. As the nut running is complete, the engine is detached from the engine plate and is placed on the adaptor plate. The gantry arm brings the engine assembly, placed on the adaptor plate, to the off-line rail and places it on the offline pallet as shown in part ‘3’ of the Figure 123. The engine assembly is detached and brought on to the off-line and is moved to the next station for packaging and final despatch as shown in part ‘4’ of the Figure 123.

Now as the four-cylinder engine is run through the existing station, as is done for the previous stages of product, the possible assembly hard points can be detected. Two amendments are required on this station i.e. adaptor plate and Y-axis gantry. The existing adaptor plate is designed to place and carry three-cylinder engine. As the engine dimensions are changed especially the increase in width shall require readjustment of resting pads on the adaptor plate, the existing adaptor plate is shown in Figure 124.

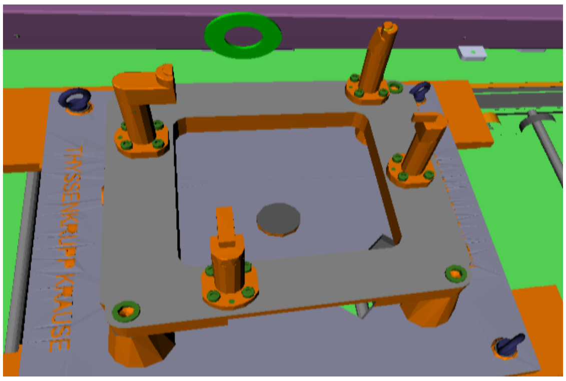


Figure 124: Adaptor Plate for OP2970

Similarly, the increase in width is exactly equal to the clearance space between the edge of the Y-axis gantry and the extreme end of the engine, therefore, for the four-cylinder engine the clearance space reduces to near zero, i.e. increase in width hinders with the Y-axis gantry, which needs to be reconfigured as well as shown in Figure 125.

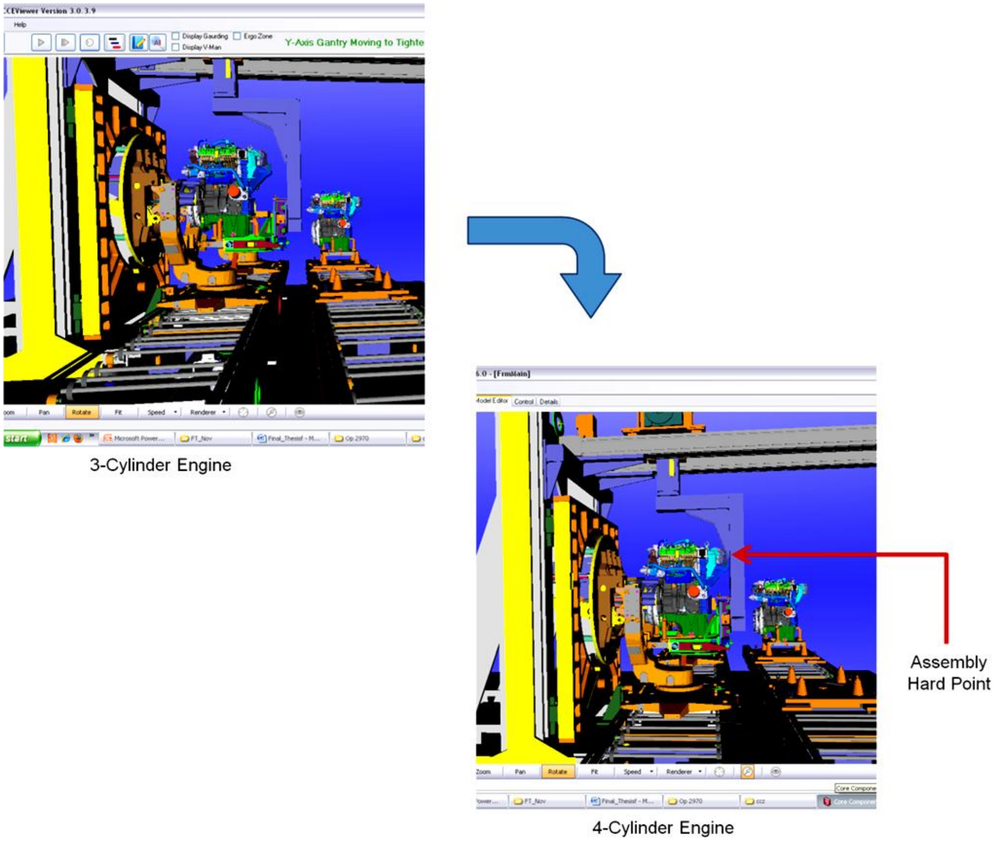


Figure 125: Existing OP2970 station limitation to off-load four-cylinder engine

The summary of the KCs, attributes and rules is provided in Table 39.

KCs	Existing attribute	New / Changed attribute	Rules / Constraints	
			Rules	For System
OP2970_Nut_Runner Qty Diameter Length Torque	4 6 20 10	4 6 20 10	If ?qty = 4 AND ((?dia > 6 and ?dia =< 25) or (?lenght !=20 and ?length < 50) or (?torque > 11) THEN Reprogram NutRunner ELSE IF ?qty != 4 or ((?dia > 25) or ?length !=20 or ?torque >11 THEN Reconfigure NutRunner ELSE IF ?qty != 4 or ((?dia > 25) or ?length !=20 or ?torque >11 THEN ReDesign NutRunner If L<450 then use as is Else If 450<L<1000 Then Reconfigure Else If L > 1000 then Redesign	
OP2970_Adaptor_Plate The plate which is used to place the engine upon Adaptor plate is affected if engine dimensions are changed		If the increase in depth of oil pan or increase in height of engine is less than 100mm then the same adaptor plate can be used	?width > 560 and ?width < 635) or (?length > 560 and ?length < 700) or (?height>640 and ?height< 875) THEN Reconfigure Adaptor Plate ELSE IF ?width > 635) OR (?length > 700) OR (?height> 875) THEN Redesign Adaptor Plate	

OP2970_Gripper Distance between jaws Distance between jaws is affected if adaptor plate is redesigned	430	430	If adaptor plate is redesigned then reprogram distance between jaws
OP2970_Gantry_Arm Distance b/w rails	200	200	

Product length x width x height
(clearance in x, y, z)
	Length = 550
	Width = 560
	Height = 640
		Length = 660
		Width =555
		Height = 630
			IF (?length=550) and (?width=560) and (?height=640) THEN Same System
			IF (?length > 560 and ?length < 635) and (?width > 560 and ?width < 700) and (?height>640 and ?height< 875) THEN ReProgram System
			ELSE IF (?length >635 OR ?width > 700 OR ?height>875) THEN Reconfigure
			ELSE IF (?length >635 AND (?width > 700 or ?height>875) THEN Redesign

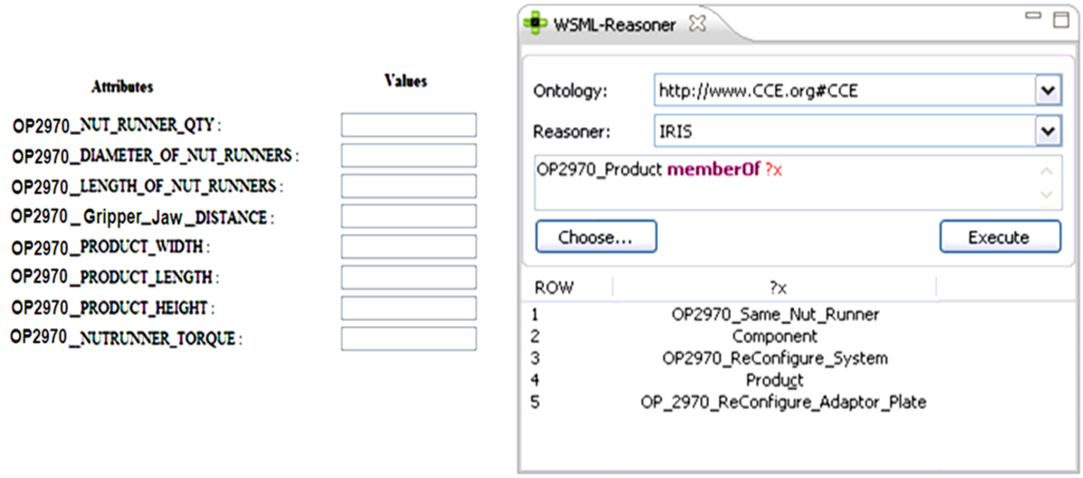
Table 39: Rules and axioms formulated on the KCs of OP2970 System

Once the rules are formulated, the next step is to provide input values for the new / changed product in terms of key characteristics of products and resources. The new / changed values of the KCs can be provided through web based form or by direct input into the WSMT editor. The KB system has got the changed parameters of the product for the ‘oil pan run down’ product build stage (product-set) to be compared with the existing parameters for the system OP2970. These two KCs can be compared through defined rules and axioms. When the query is run, the system has got both the set of parameters (existing vs new) and it will compare the two values for each KC defined and provide the output, e.g. if the user wants to know the effects of the changed product on the existing system, the results may be obtained as shown below in Table 40.

OP2970 (RTV Station)		
Input (working memory)		**Output**
OP2970_Nut_Runner_Qty	4	OP2970 Same Nut Runner OP2970 Reconfigure Adaptor Plate OP2970 Reconfigure System
OP2970_Nut_Runner_Diameter	6	
OP2970_Nut_Runner_Length	20	
OP2970_Nut_Runner_Torque	<11	
OP2970_Gripper_Jaw_Distance (Gantry Arm)	430	
OP2970_Product_Length	660	
OP2970_Product_Width	555	
OP2970_Product_Height	630	

Table 40: Summary of Inputs and outputs for OP2970 in ALDIMS

A snapshot of input parameters and output from the ALDIMS is shown below in Figure 126.



Attributes	Values
OP2970_NUT_RUNNER_QTY :	
OP2970_DIAMETER_OF_NUT_RUNNERS :	
OP2970_LENGTH_OF_NUT_RUNNERS :	
OP2970_Gripper_Jaw_DISTANCE :	
OP2970_PRODUCT_WIDTH :	
OP2970_PRODUCT_LENGTH :	
OP2970_PRODUCT_HEIGHT :	
OP2970_NUTRUNNER_TORQUE :	

WSML-Reasoner

Ontology: http://www.CCE.org#CCE

Reasoner: IRIS

OP2970_Product memberOf ?x

ROW	?x
1	OP2970_Same_Nut_Runner
2	Component
3	OP2970_ReConfigure_System
4	Product
5	OP_2970_ReConfigure_Adaptor_Plate

Figure 126: Input KCs and the query output result by the KB system for OP1900

As shown in the query output, the ALDIMS tool suggests reconfiguring the system and adaptor plate. This was verified virtually and found in perfect harmony to the detailed simulation results. Therefore, the changed product affecting different stations along specific assembly operations also affects the last station i.e. engine off-load station. As the depth of oil pan sump is increased, it was initially envisaged that this might affect the engine adaptor plate, however, the engine adaptor plate is solid along the edges and hollow in the centre (i.e. a through square gap in the centre) therefore, the increased depth of the engine still can be handled unaffected on this station. In this way effects of change on a few stations, on a zone or on the whole of the assembly line can be evaluated quickly and confidently.