IBM Software

UKInnovate2010

The Rational Software Conference

Smarter software for a smarter planet.

IBM Software

UKInnovate2010

The Rational Software Conference

Using Model Driven Development for Smarter Design

Andy Bentley
IBM GBS UK

Smarter software for a smarter planet.

Introductions and Agenda

IBM UK GBS

Smarter Design

- Why is it needed?
- How we used it
- Why is it “Smart”?

andrew.n.bentley@uk.ibm.com

Abstract

This presentation will share how the team dealt with challenging timescales and tight budgets by:

Finding a hidden IBM nugget,

Working with Software Group to understand its capability

Developing a reusable asset that to exploit it which resulted in a **25% saving** in projected cost for Design, Build and Test

The asset enabled **early project delivery** with **zero production defects** and led to **soaring client satisfaction** - *"it's the best project we've ever had here".*

"This asset saved our bacon" – IBM Delivery Exec

Smarter Planet

Smart ideas

KERS

E-Boarding passes

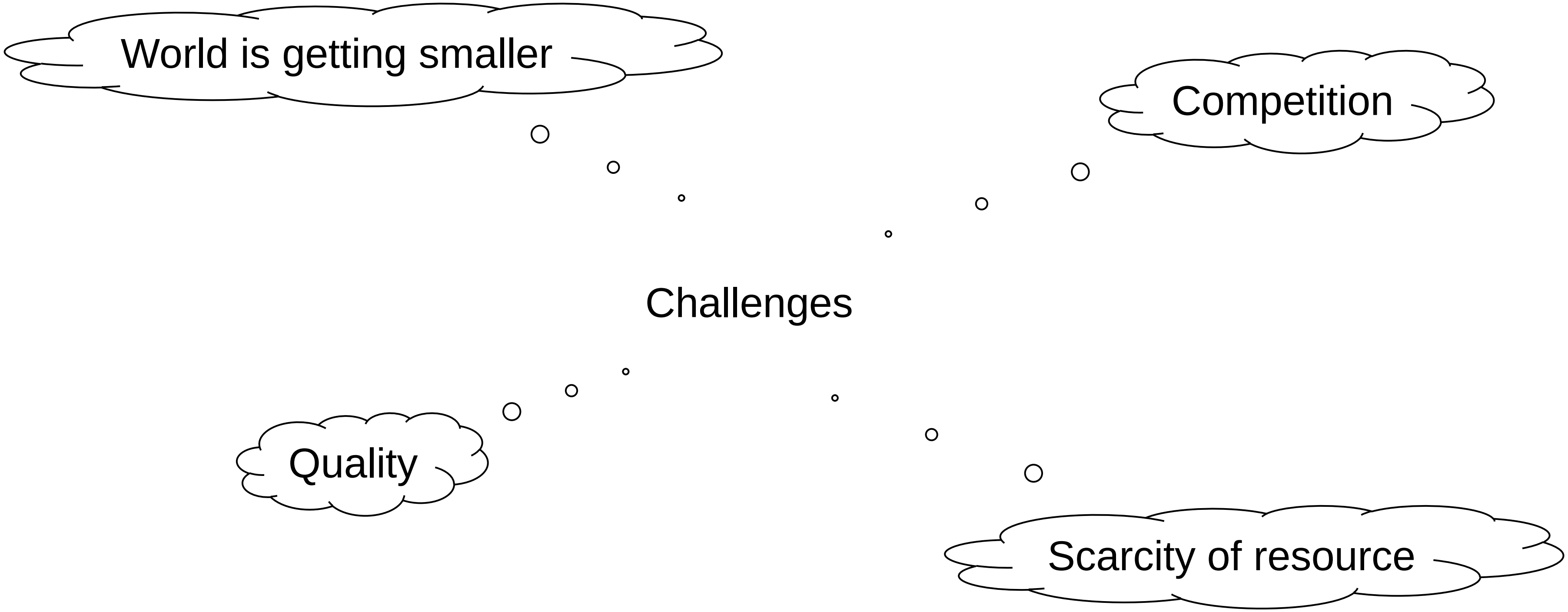
Predictive actions e.g. you’re about to get a flat tyre

Continuous safety checking – the dam has too much pressure

Dashboards – light in the study

Pay as you drive insurance

Why do we need Smarter Design



World is getting smaller

Competition

Challenges

Quality

Scarcity of resource

My Situation

Fixed price – "sales guy pricing"

New team

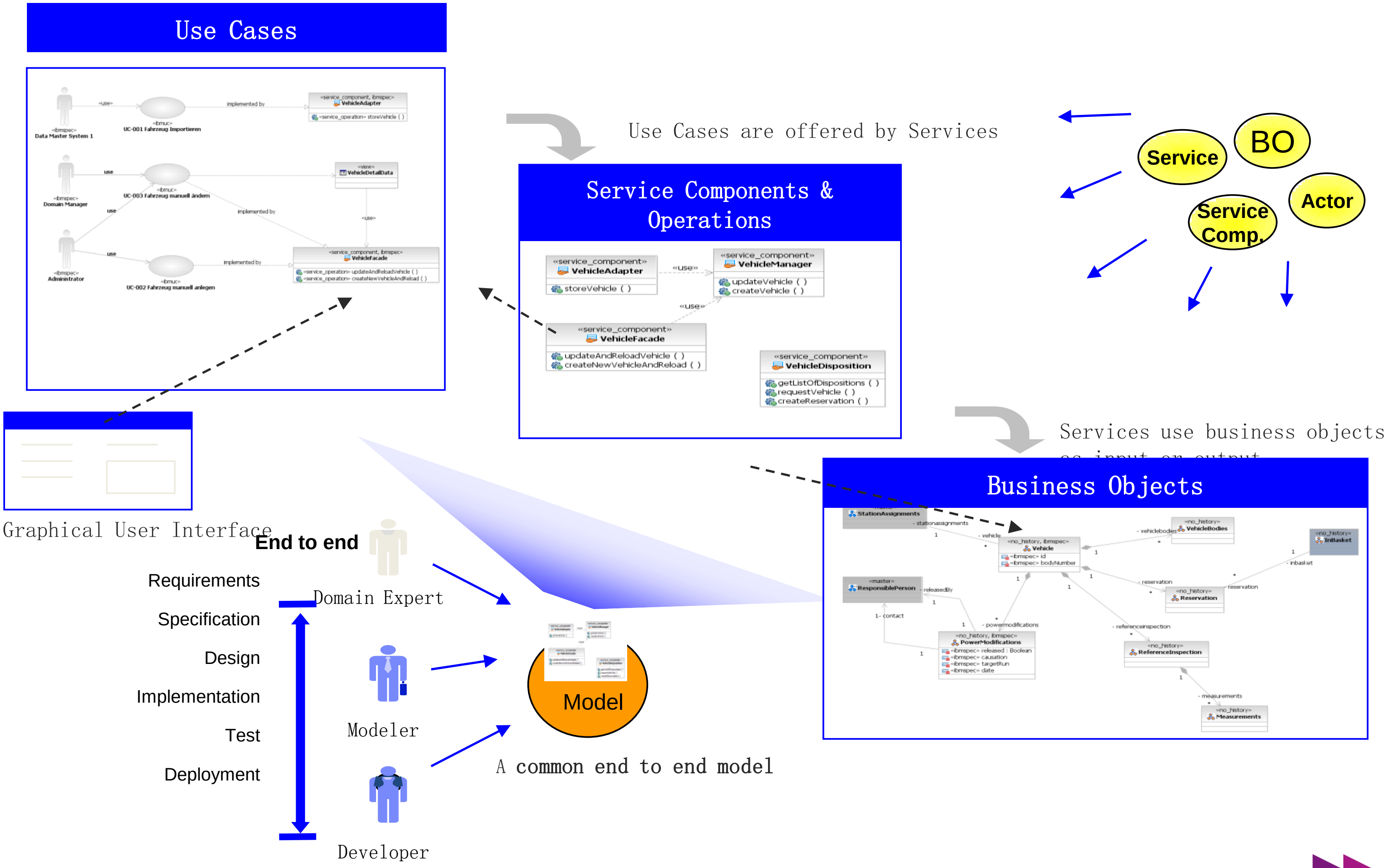
Delivery Model

- Onshore
- Landed
- offshore

Other challenges

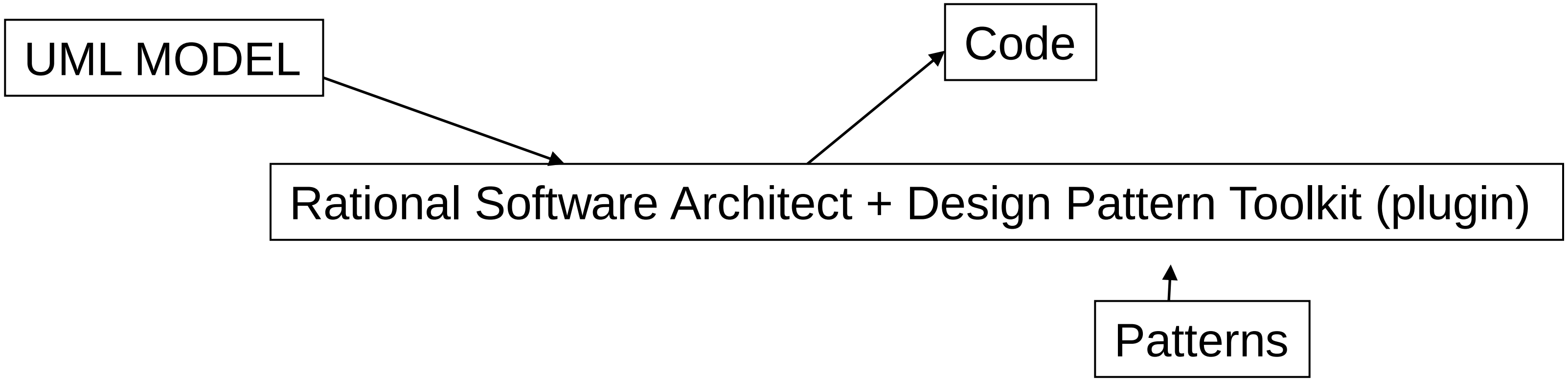
- Competitors
- Quality
- Need to gain more work

Classic Software modelling and development approach


Use Cases
«use»
implemented by
«service_component, ibmspec» VehicleAdapter
«service_operation» storeVehicle ( )
«ibmspec» Data Master System 1
«ibmuc» UC-001 Fahrzeug Importieren
use
«view» VehicleDetailData
«ibmspec» Domain Manager
«ibmuc» UC-003 Fahrzeug manuell ändern
use
implemented by
«use»
«ibmspec» Administrator
use
«ibmuc» UC-002 Fahrzeug manuell anlegen
implemented by
«service_component, ibmspec» VehicleFacade
«service_operation» updateAndReloadVehicle ( )
«service_operation» createNewVehicleAndReload ( )
Use Cases are offered by Services
Service Components & Operations
«service_component» VehicleAdapter
storeVehicle ( )
«use»
«service_component» VehicleManager
updateVehicle ( )
createVehicle ( )
«use»
«service_component» VehicleFacade
updateAndReloadVehicle ( )
createNewVehicleAndReload ( )
«service_component» VehicleDisposition
getListOfDispositions ( )
requestVehicle ( )
createReservation ( )
Service
BO
Service Comp.
Actor
Graphical User Interface
Services use business objects as input or output
Business Objects
«master» StationAssignments
- stationassignments
1
- vehicle
*
«no_history, ibmspec» Vehicle
«ibmspec» id
«ibmspec» bodyNumber
- vehiclebodies
«no_history» VehicleBodies
«no_history» InBasket
- inbasket
- reservation
«no_history» Reservation
- reservation
«master» ResponsiblePerson
- releasedBy
1- contact
- powermodifications
«no_history, ibmspec» PowerModifications
«ibmspec» released : Boolean
«ibmspec» causation
«ibmspec» targetRun
«ibmspec» date
- referenceinspection
«no_history» ReferenceInspection
- measurements
«no_history» Measurements
End to end
Requirements
Specification
Design
Implementation
Test
Deployment
Domain Expert
Modeler
Developer
Model
A common end to end model

Connecting the dots

“If you know what your model is and you know what the output needs to look like you can create a patter to connect the XMI based model to Java Code”


UML MODEL
Code
Rational Software Architect + Design Pattern Toolkit (plugin)
Patterns

We focused on producing high quality, robust, efficient, well documented and standards compliant code

What are Patterns and Assets?

[Industry Definition] **Patterns** are solutions to recurring problems in a given context.

Assets are artifacts with business value

[IBM] A **Pattern Description** is an asset that allows a Pattern to be communicated between humans, and may be described using many asset types, e.g. structured text, diagrams, and code samples

[IBM] A **Pattern Implementation** is an asset which automates the application of a pattern in a particular environment, eg RSA Pattern, RSA Transformation, Plug-in, DPTK pattern.

[ISSW] An **Exemplar** is a set of artifacts that conforms to best practices, conventions, and guidelines within a given context, and from which a pattern implementation can be extracted

How did the tooling help

We actually started with WASD before moving to RSA

Created 23 “accelerators”

Removed the “silly error”

Coders concentrate on business logic

It allowed an automatic, repeatable approach

With the use of the code templates, best-practice can be easily applied

Reduced project risk

Concurrent and iterative development across local and global teams

A quick look under the covers

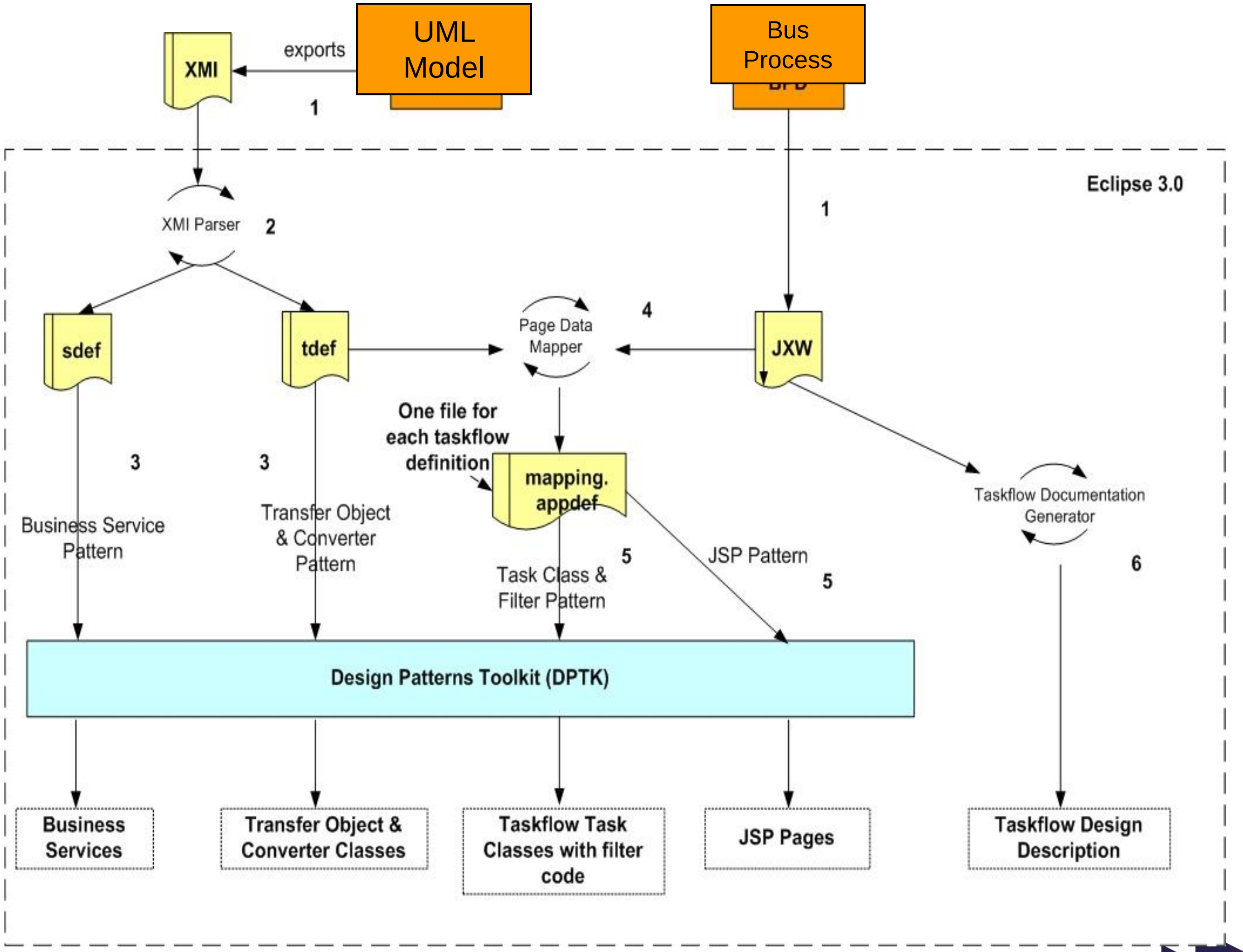
We created four independent Eclipse plug-ins:

- Service
- JXW documentation
- Tasks
- Page Data Mapper

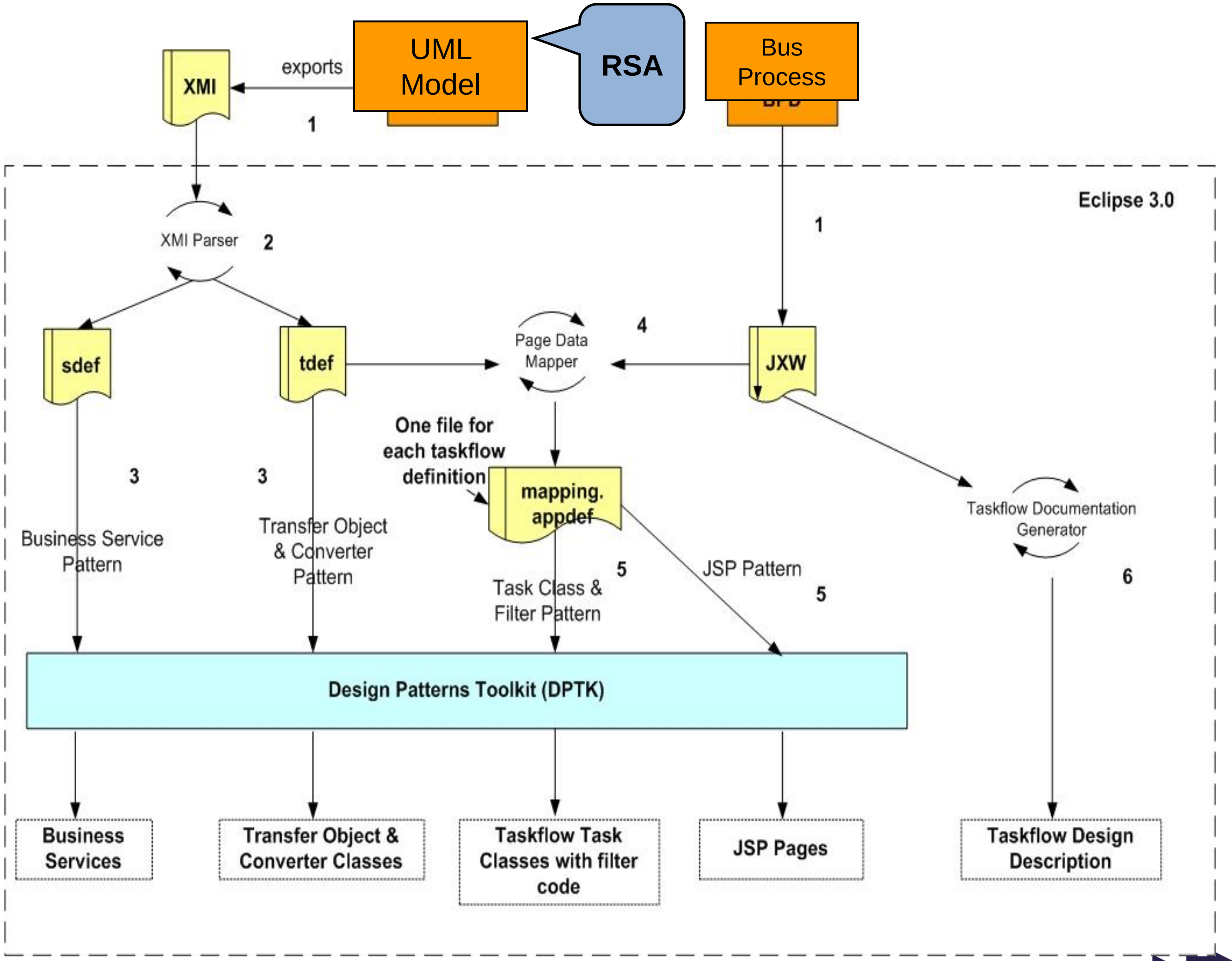
Benefits:

- It ensured that designers/developers had a very intuitive interface to generating code
- It allowed the Designer/Developer to stay within Rational Software Architect throughout the design, build and unit test phases of the project.

Process for the Designer/Developer


UML Model
Bus Process
XMI
exports
1
Eclipse 3.0
1
XMI Parser
2
4
sdef
tdef
Page Data Mapper
JXW
One file for each taskflow definition
mapping. appdef
3
3
Business Service Pattern
Transfer Object & Converter Pattern
Task Class & Filter Pattern
5
JSP Pattern
5
Taskflow Documentation Generator
6
Design Patterns Toolkit (DPTK)
Business Services
Transfer Object & Converter Classes
Taskflow Task Classes with filter code
JSP Pages
Taskflow Design Description

Process for the Designer/Developer



UML Model

RSA

Bus Process

exports

1

XMI

Eclipse 3.0

1

XMI Parser

2

sdef

tdef

Page Data Mapper

4

JXW

One file for each taskflow definition

mapping. appdef

Taskflow Documentation Generator

3

3

Business Service Pattern

Transfer Object & Converter Pattern

5

JSP Pattern

5

Task Class & Filter Pattern

6

Design Patterns Toolkit (DPTK)

Business Services

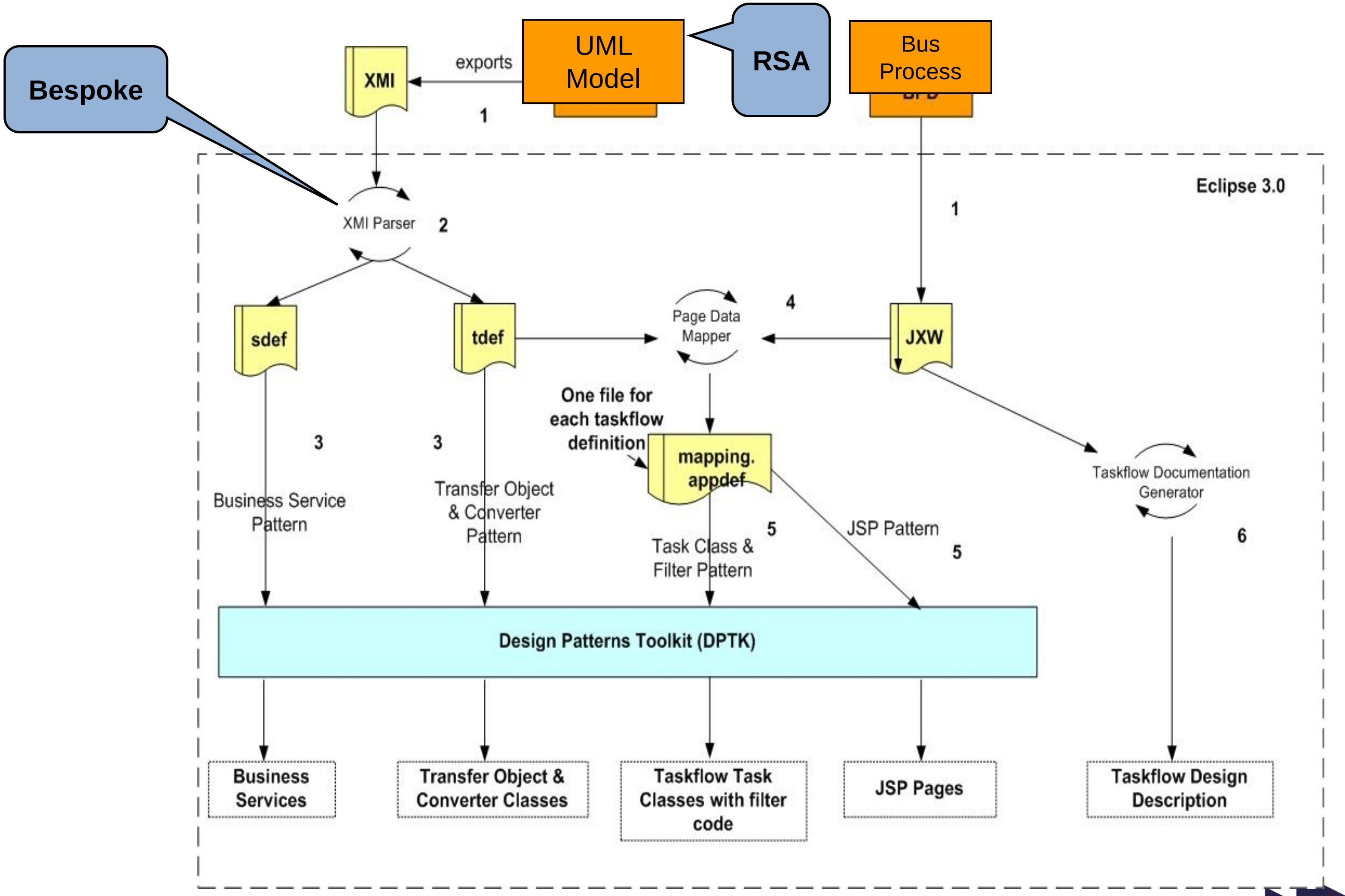
Transfer Object & Converter Classes

Taskflow Task Classes with filter code

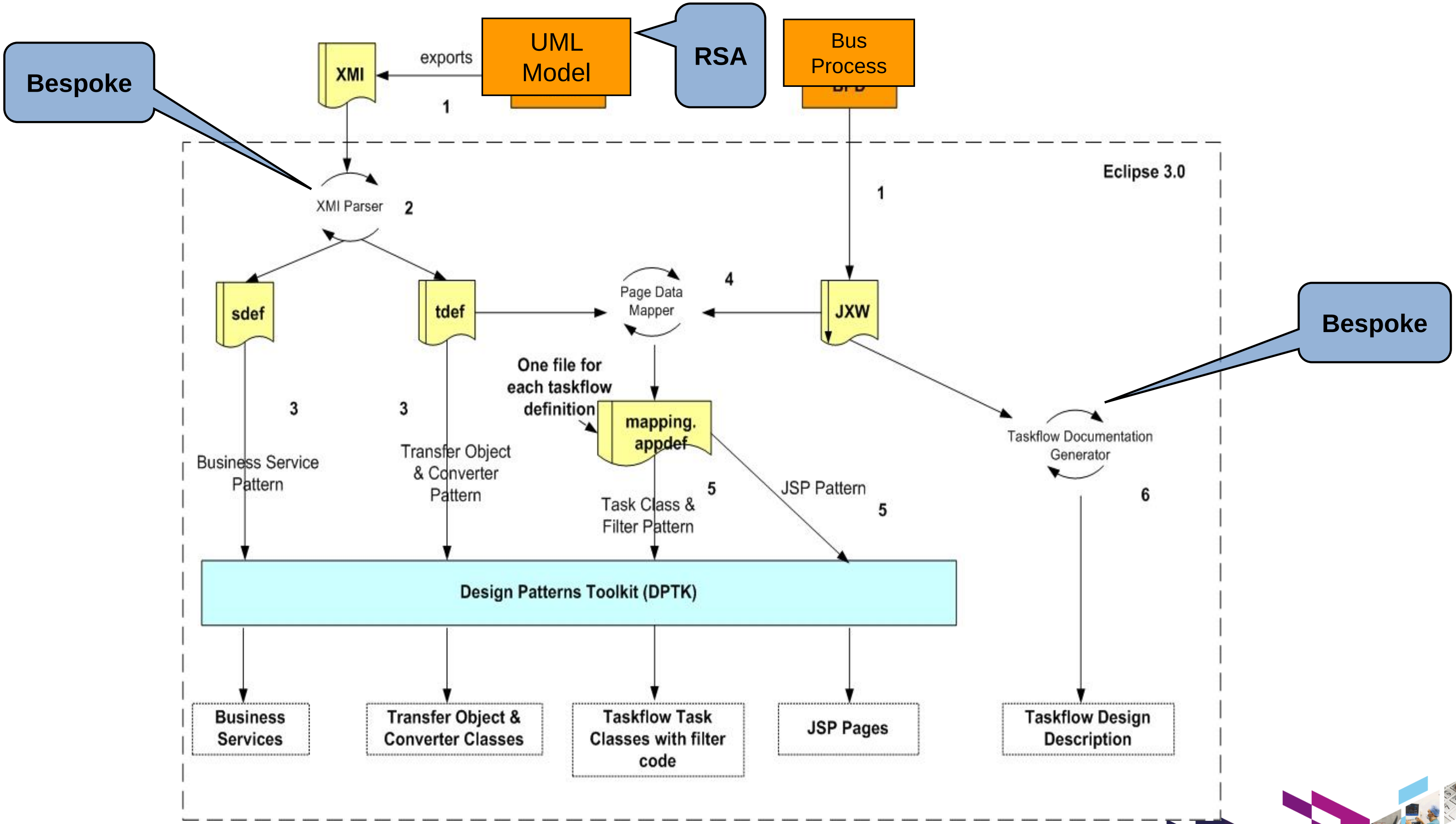
JSP Pages

Taskflow Design Description

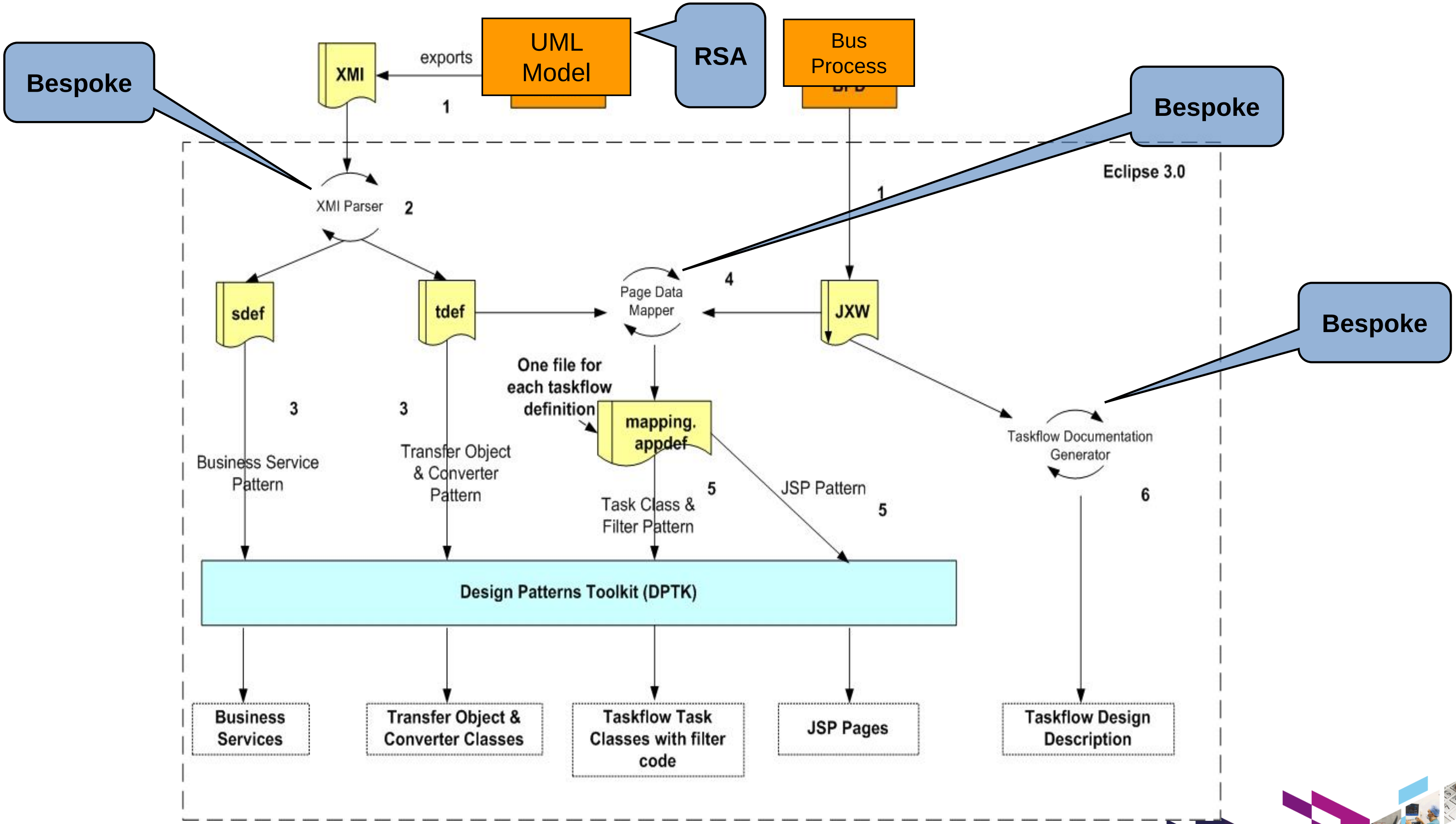
Process for the Designer/Developer


Bespoke
XMI
exports
UML Model
RSA
Bus Process
BPD
1
Eclipse 3.0
XMI Parser
2
1
sdef
tdef
Page Data Mapper
4
JXW
One file for each taskflow definition
mapping. appdef
3
3
Business Service Pattern
Transfer Object & Converter Pattern
Task Class & Filter Pattern
5
JSP Pattern
5
Taskflow Documentation Generator
6
Design Patterns Toolkit (DPTK)
Business Services
Transfer Object & Converter Classes
Taskflow Task Classes with filter code
JSP Pages
Taskflow Design Description

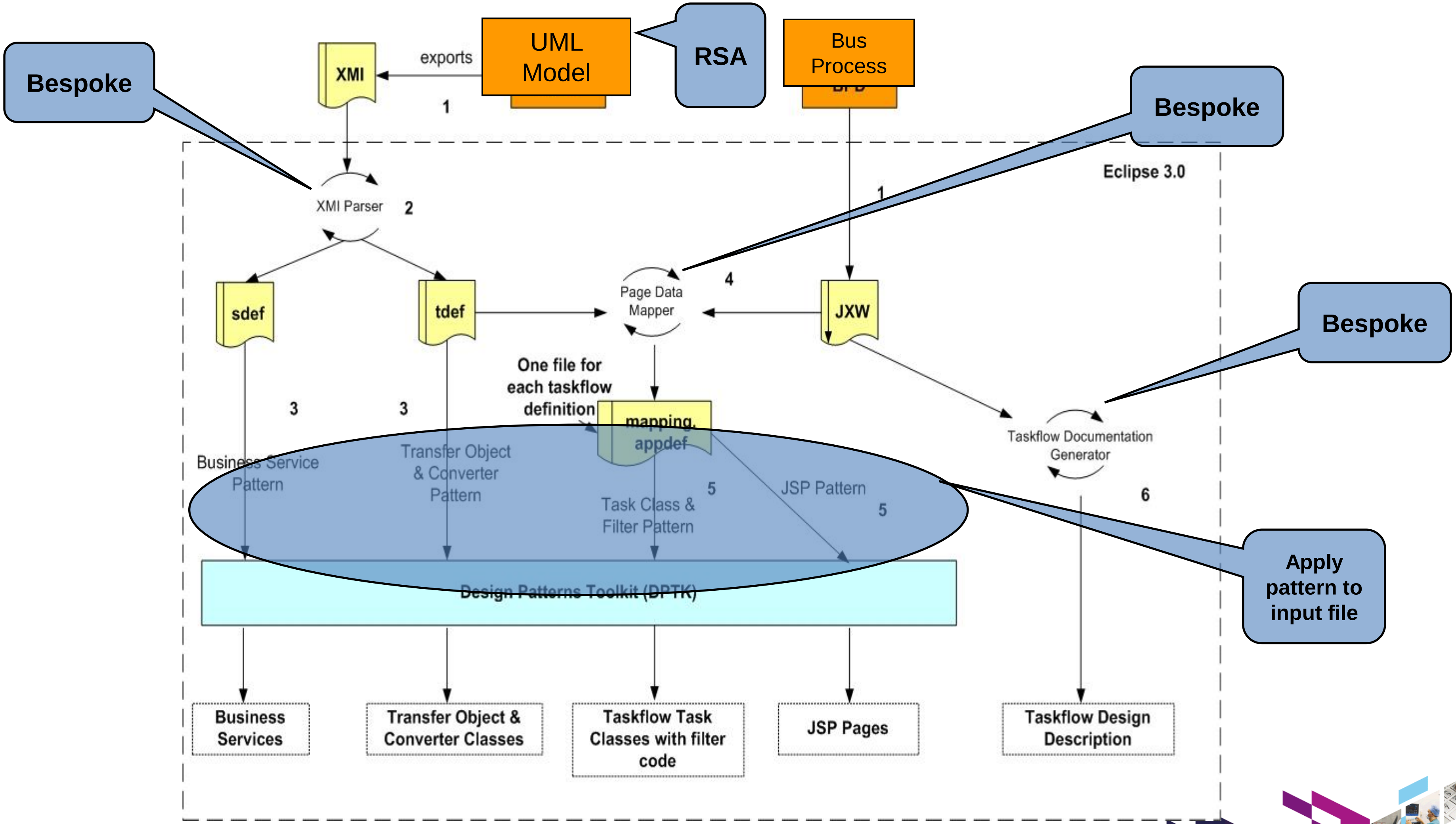
Process for the Designer/Developer


Bespoke
XMI
exports
UML Model
1
RSA
Bus Process
Eclipse 3.0
1
XMI Parser
2
sdef
tdef
Page Data Mapper
4
JXW
Bespoke
One file for each taskflow definition
mapping.appdef
3
3
Business Service Pattern
Transfer Object & Converter Pattern
Task Class & Filter Pattern
5
JSP Pattern
5
Taskflow Documentation Generator
6
Design Patterns Toolkit (DPTK)
Business Services
Transfer Object & Converter Classes
Taskflow Task Classes with filter code
JSP Pages
Taskflow Design Description

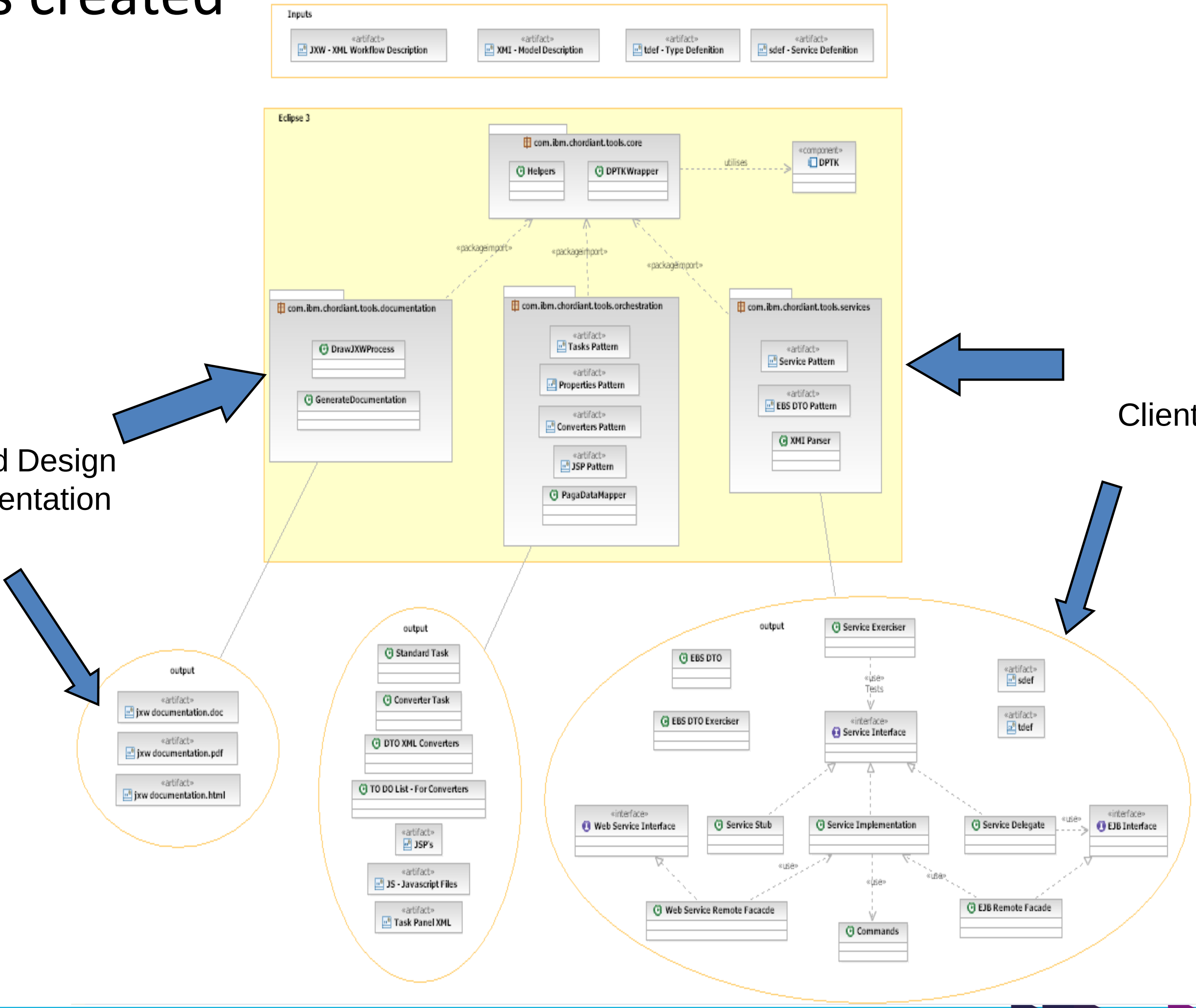
Process for the Designer/Developer


Bespoke
XMI
exports
UML Model
RSA
Bus Process
BPD
1
Bespoke
Eclipse 3.0
XMI Parser
2
1
Page Data Mapper
4
sdef
tdef
JXW
Bespoke
One file for each taskflow definition
mapping. appdef
3
3
Business Service Pattern
Transfer Object & Converter Pattern
Task Class & Filter Pattern
5
JSP Pattern
5
Taskflow Documentation Generator
6
Design Patterns Toolkit (DPTK)
Business Services
Transfer Object & Converter Classes
Taskflow Task Classes with filter code
JSP Pages
Taskflow Design Description

Process for the Designer/Developer


Bespoke
XMI
exports
UML Model
1
RSA
Bus Process
Bespoke
Eclipse 3.0
1
XMI Parser
2
4
sdef
tdef
Page Data Mapper
JXW
Bespoke
One file for each taskflow definition
3
3
mapping, appdef
Business Service Pattern
Transfer Object & Converter Pattern
Task Class & Filter Pattern
5
JSP Pattern
5
Taskflow Documentation Generator
6
Apply pattern to input file
Design Patterns Toolkit (DPTK)
Business Services
Transfer Object & Converter Classes
Taskflow Task Classes with filter code
JSP Pages
Taskflow Design Description

Objects created

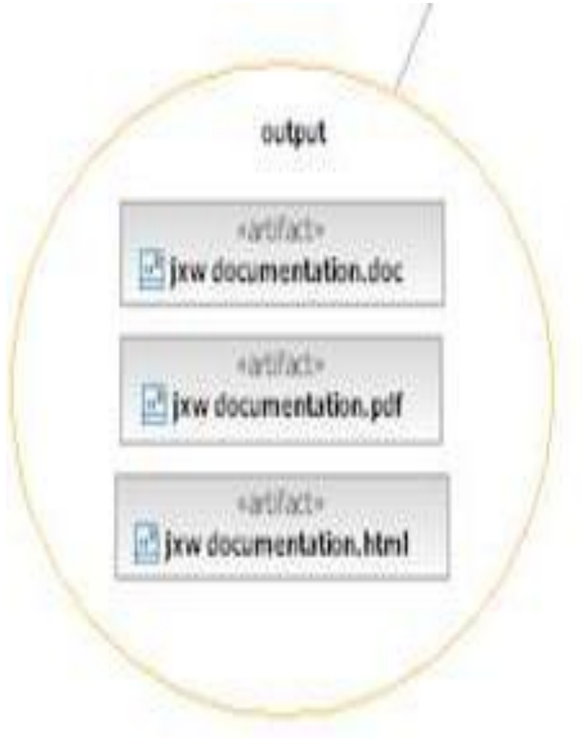

Inputs
«artifact» JXW - XML Workflow Description
«artifact» XMI - Model Description
«artifact» tdef - Type Defenition
«artifact» sdef - Service Defenition
Eclipse 3
com.ibm.chordiant.tools.core
Helpers
DPTKWrapper
utilises
«component» DPTK
«packageimport»
«packageimport»
«packageimport»
com.ibm.chordiant.tools.documentation
DrawJXWProcess
GenerateDocumentation
com.ibm.chordiant.tools.orchestration
«artifact» Tasks Pattern
«artifact» Properties Pattern
«artifact» Converters Pattern
«artifact» JSP Pattern
PagaDataMapper
com.ibm.chordiant.tools.services
«artifact» Service Pattern
«artifact» EBS DTO Pattern
XMI Parser
output
«artifact» jxw documentation.doc
«artifact» jxw documentation.pdf
«artifact» jxw documentation.html
output
Standard Task
Converter Task
DTO XML Converters
TO DO List - For Converters
«artifact» JSP's
«artifact» JS - Javascript Files
«artifact» Task Panel XML
output
Service Exerciser
EBS DTO
EBS DTO Exerciser
«use» Tests
«interface» Service Interface
«artifact» sdef
«artifact» tdef
«interface» Web Service Interface
Service Stub
Service Implementation
Service Delegate
«use»
«interface» EJB Interface
«use»
«use»
«use»
Web Service Remote Facacde
Commands
EJB Remote Facade

Detailed Design Documentation

Client Specific

Under the covers – Workflow Documentation artefact

- Taskflow design document
  - Detailed taskflow diagrams
  - All associated descriptions and comments included
  - All tasks within the taskflow and associated Java implementation classes listed and described
  - Links dynamically to associated Javadoc (html version) for the Java implementation classes
  - Can be generated in MS Word, Adobe PDF or HTML format



Under the covers – Task, JSP & XML Converter artefacts

Task classes

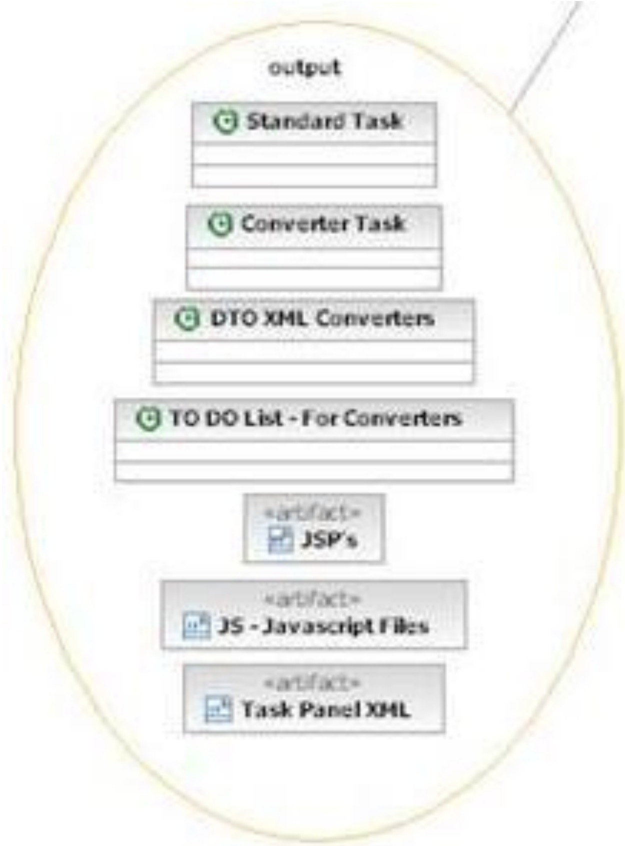
- Standard tasks are templates which simply need to have logic filled in a well-defined area manually
- Converter tasks are generated in their entirety, require no modification manually

JSP pages

- All pages defined for the solution are generated as templates
- Each page has a pre-defined structure, including common pieces of visual components, common Javascript imports and style sheet references
- Simply need the page content filled in manually

XML Converters

- Generated to aid the performance of the Chordiant application in marshalling from Object to XML and vice versa
- Entirely generated, no need to modify manually



Under the covers – Business service tier artefacts

Data transfer objects

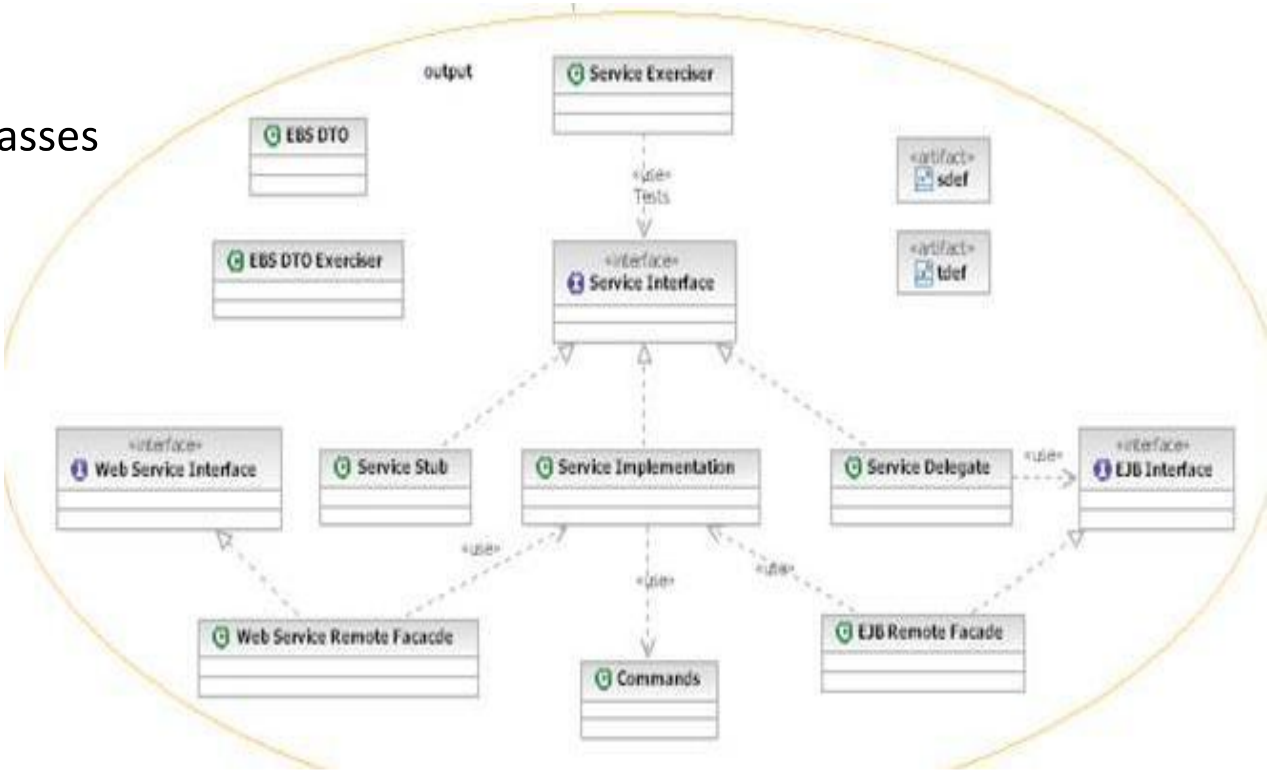
Service classes

- Core implementation class
- Command classes
- Service interface

Distributed deployment classes

- Service delegate
- EJB interface
- EJB Remote Façade (stateless)
- Web service remote façade
- Web service interface

- Testing classes
  - Service exerciser
  - Service stub



Initial results

- Decreased cost of delivery
  - 140,000 lines of generated code, 70% of total
  - 2 weeks ahead of 6 month schedule
  - To budget
- Defects attributable to generated code = 2
  - Both were introduced by people not following the process
- Most important of all
  - Very high client satisfaction - > more work -> factory approach->more offshore -> reduced costs and prices

Why was this "Smart"?

Patterns provide a powerful way to improve product development by identifying best practices and design expertise that can be captured in tools and are then available for reuse.

By using patterns, your organisation can:

Improve productivity
Reduce development time
Minimize complexity
Increase quality
Improve governance
Gain business agility
Leverage IT skills
Promote open standards
Close the gap between business and IT
Improve cost

Innovation re-use

3 other UK clients

In Germany

- 1.2 million LoC generated out of a total of 1.5 million LoC

Patterns Initiative

- http://www.ibm.com/developerworks/rational/products/patternsolutions/index.html

Summary

For me the innovation provided Unique Selling Proposition based on

Quality

Rigour

Repeatability

Facilitated lower risk when working offshore

Use of these accelerators were factored into IBM’s successful proposal for the following project allowing us to reduce our estimates by 25%.

Other applications

Documentation

- Multiple formats, different languages

Release materials

- Scripts, installation packs

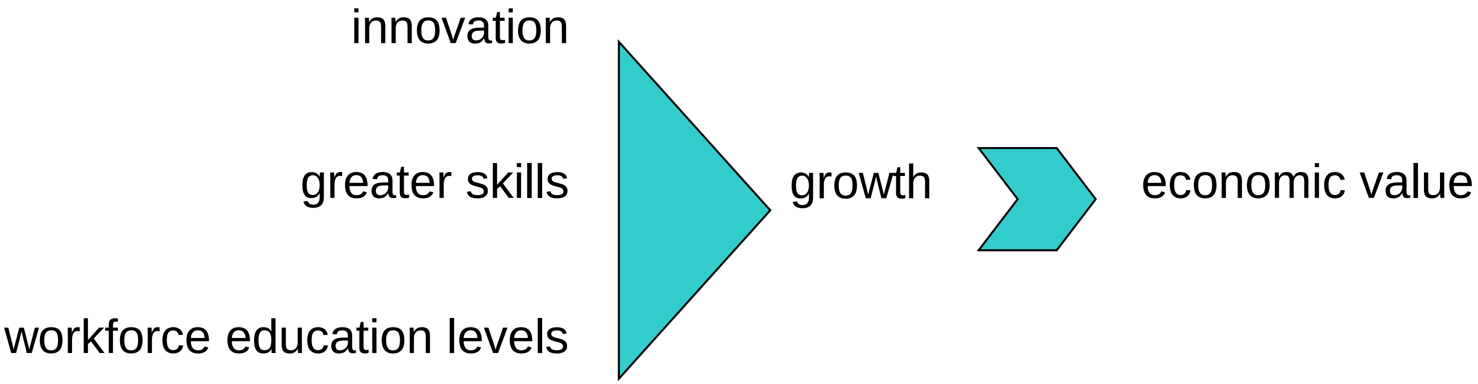
Configuration

- Generate machine configuration and procurement notes

Workflow

- Human and Machine tasks

Summary



innovation

greater skills

workforce education levels

growth

economic value

Instead of technology innovation serving primarily as a strong ***substitute*** for labour, it is becoming a ***supportive enabler*** for people and their skills, knowledge and creativity.

Final Message – It’s not a Silver Bullet

It’s a bit like the old shoemaker who designs the shoes and prepares the materials who on returning in the morning finds the work all done.

Links

IBM Redbooks

Alphaworks

Developerworks

Catalogue of patterns is available through the developerWorks pattern repository

Thank You

&

Questions?

IBM Software

UKInnovate2010

The Rational Software Conference

Smarter software for a smarter planet.