

Model-driven Engineering for the social enterprise

John Grundy

Dean, Software & Electrical Engineering

SWIN
BUR
NE

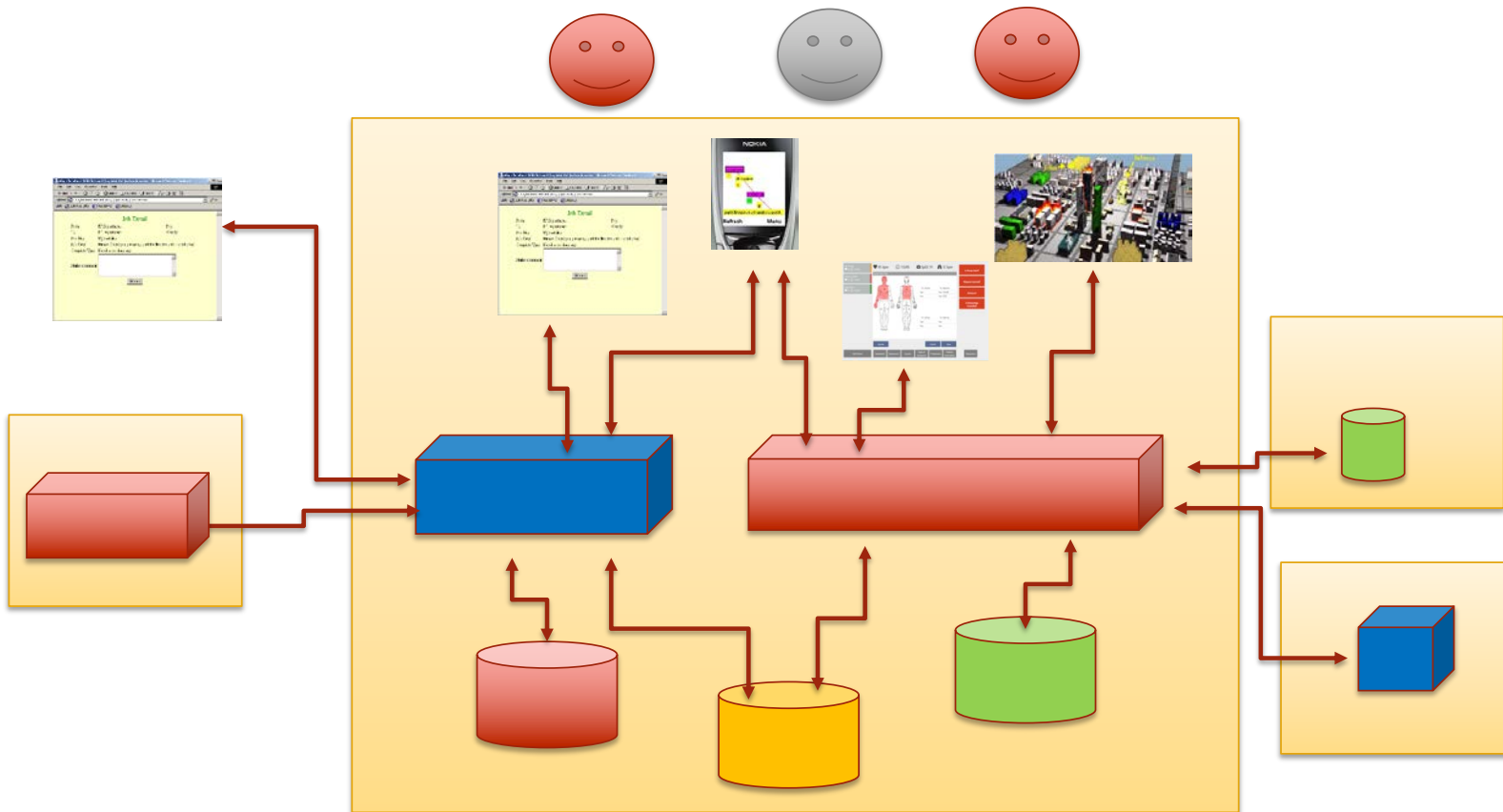
SWINBURNE
UNIVERSITY OF
TECHNOLOGY

Swinburne
▶ think forward



- The emerging social enterprise (John G interpretation anyway...)
- Example domains and data
- Requirements for next-generation social enterprise systems
- Engineering next-generation social enterprise systems
 - Modelling
 - Generating
- Some of our representative current example projects
- Future work

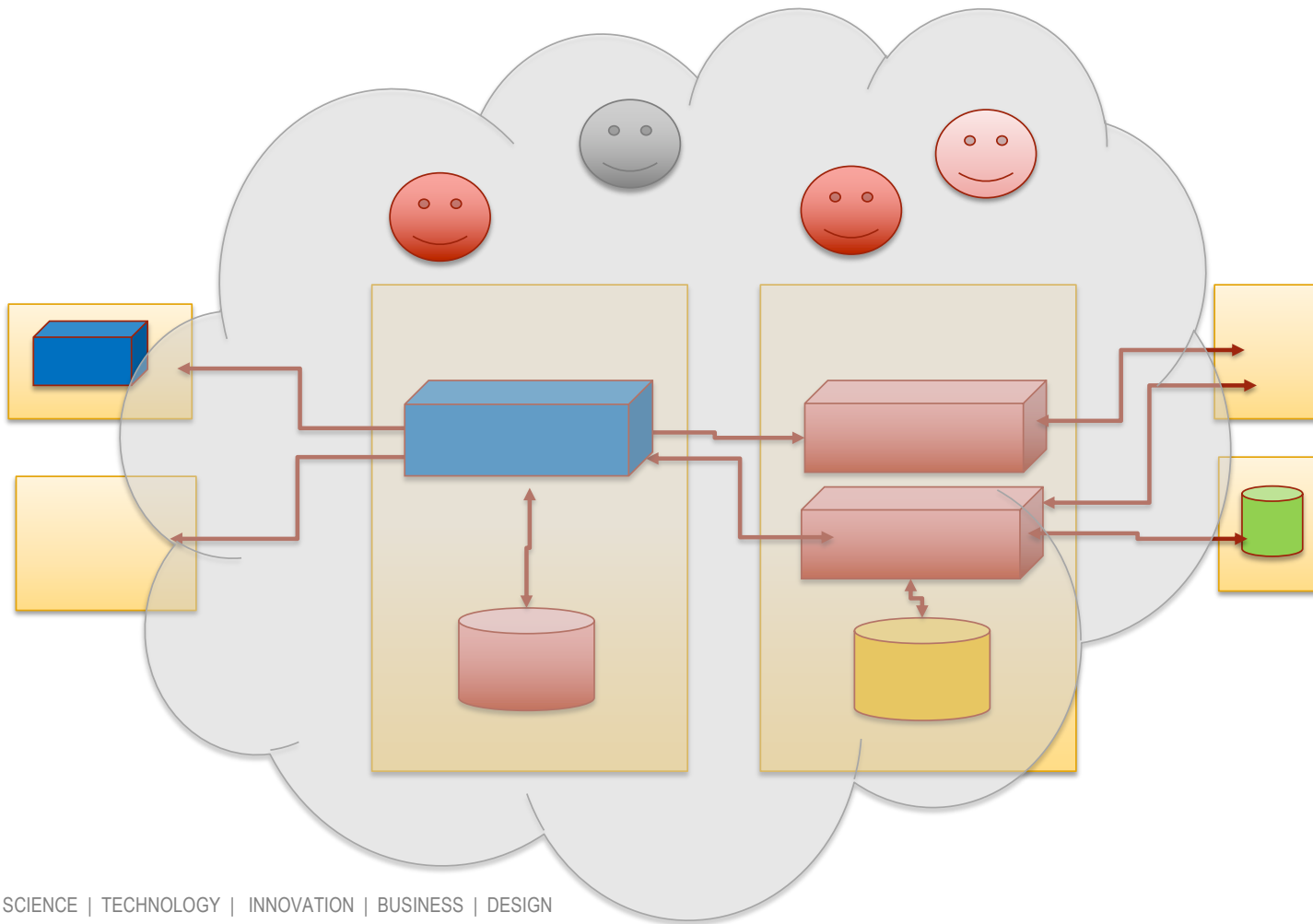
“Old-style” Enterprise Systems



- Mainly internally focused (even SOA & cloud-based!)
- Significant external data, services – but limited sources / sinks
- Enterprise to enterprise
- Transactional – B2B

Emerging Social Media Systems

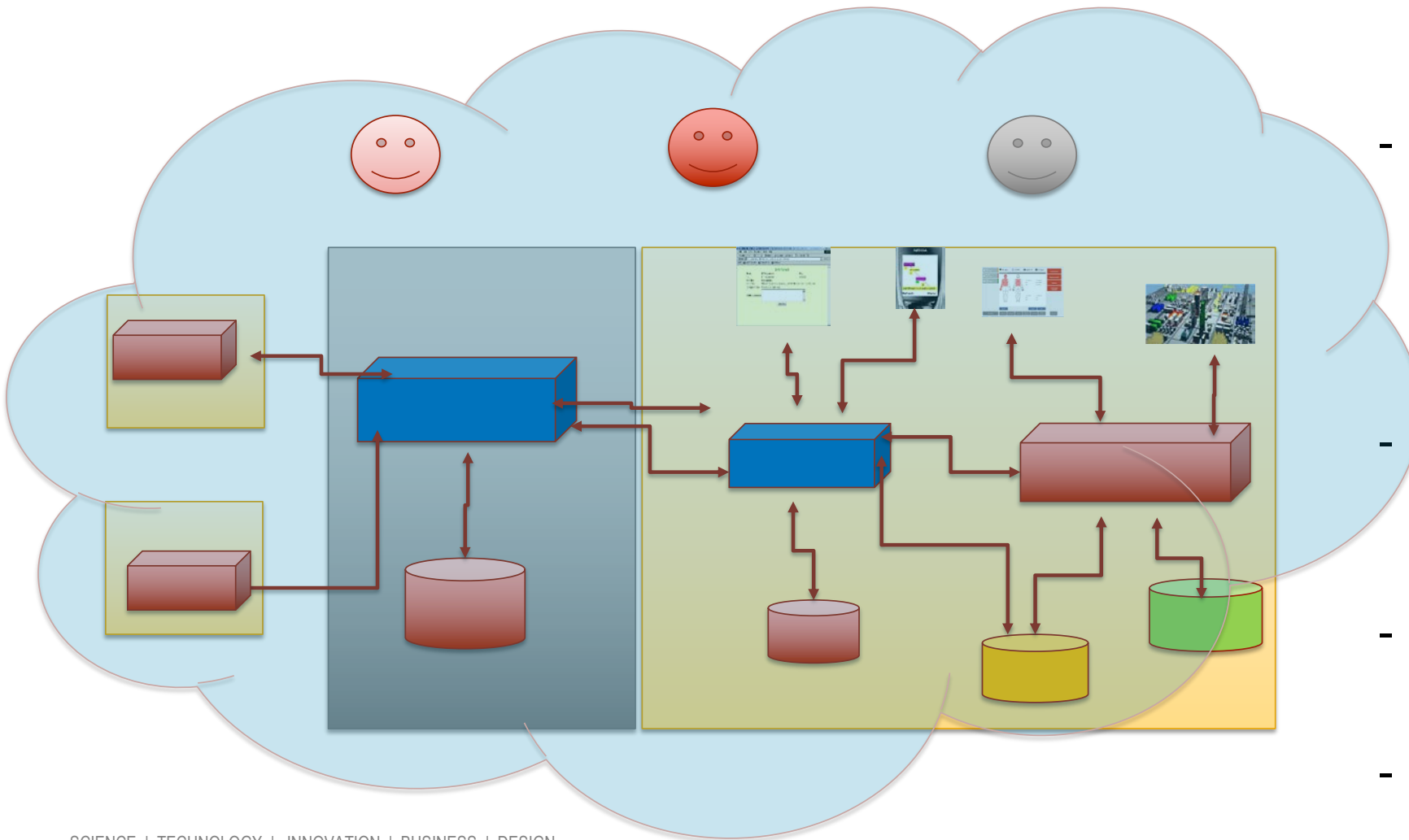
Swinburne



- Incorporate variety of social vs corporate data
- (Somewhat) limited cross-system sharing
- Increasing use in corporate domains
- Trend to integrated access
- P2P

Emerging Social Enterprise Systems

Swinburne



- To the user – where does enterprise vs social media begin / end?
- Ubiquitous enterprise and social media
- Increasing IoT/device info
- P2B



- Probably just about anything 😊, but increasingly...
- Retail – crowdsourced requirements, design, usage, reviews of products
- Financial services – micro-finance, tailored packages, on-demand services
- Government services – transport, infrastructure esp demand-based supply – personal, group, demographic data etc
- eHealth – wearable data, personalised solutions eg fitness, dietary, treatment
- Education – learning analytics merging LMS, OLM, group interaction, behavioural data



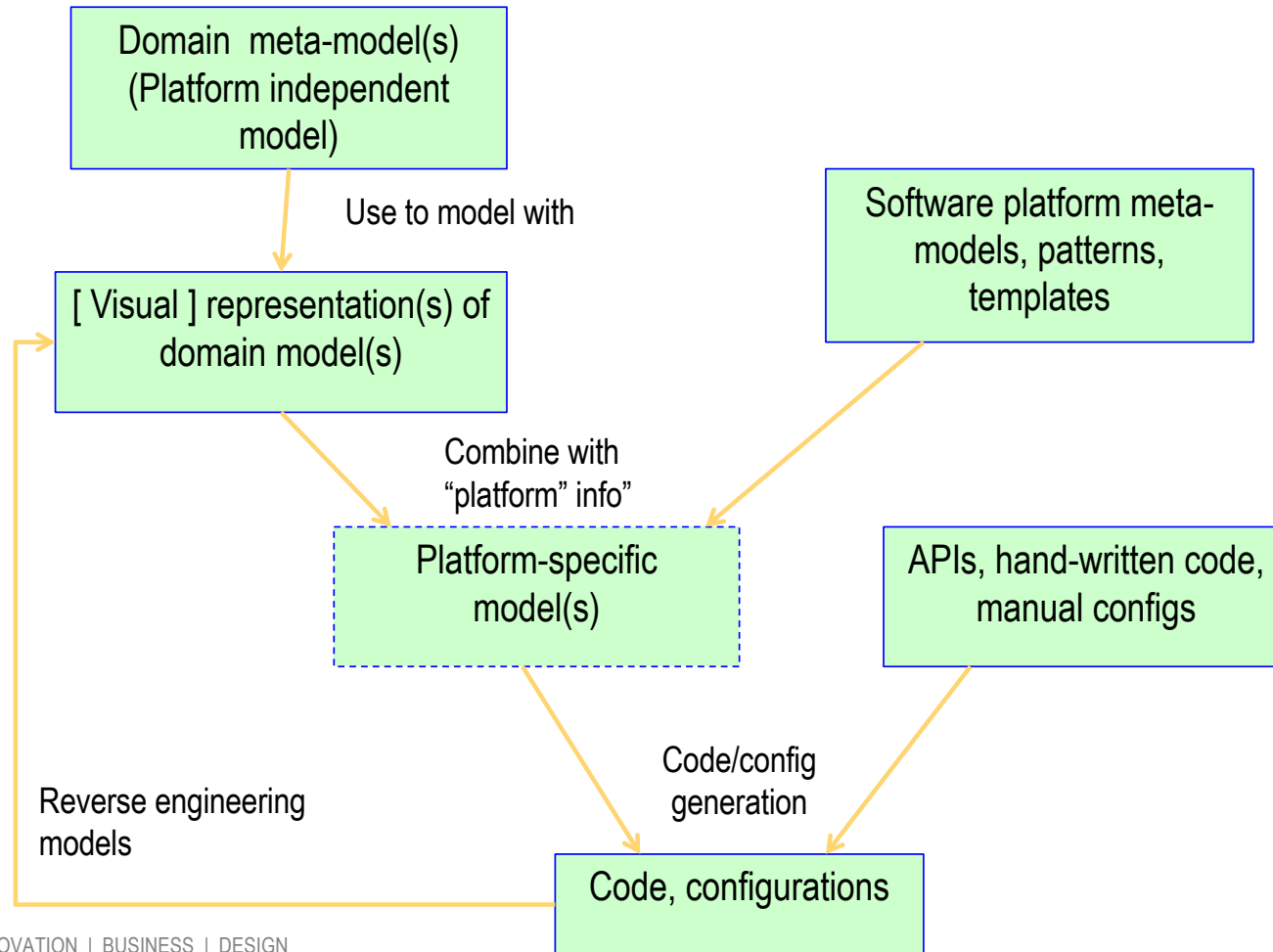
- Enterprise databases (don't forget these are still very rich!)
- eGovernment data sources
- Cloud data sources – yours and theirs
- Social media
- Crowdsourcing
- Internet of Things
 - Environment e.g. building, transport, infrastructure
 - Groups of people e.g. rooms, devices, video, voice
 - Personal data e.g. wearables etc

Key requirements – and challenges!

Swinburne



- Source (and sink) data from very diverse places
- Integrate disparate data sources
- Model complex applications
- Model and apply various data analytics
- Visualise complex information
- Realize and scale complex applications
- Maintain privacy and security
 - Where is the boundary between self / friends / enterprise / community ...





- Model disparate data sources – structured, semi-structured, discrete, continuous, number/text/image/sound/video/...
- Source data from disparate sources – enterprise systems (services), devices, social media, wide variety of formats ; some limitations for privacy etc
- Integrate / wrangle / cleanse / transform / store data
- Apply various analytical techniques to discover information / knowledge – aggregate, disaggregate, data mine, ML, feedback loops, ...
- Visualise data and information to support decisioning (by multiple people)
- Model non-functional issues: security, privacy, reliability, interfaces, ...



- Generate scripts, code to cleanse, wrangle, integrate data to produce large data sets for social enterprise data
- Generate data analytics code and/or 3rd party code configurations to apply to large data sets
- Generate apps, configurations for mobile, web and increasingly IoT-based social enterprise interfaces, including decisioning support via rich information visualisations
- Generate configurations, scripts, code to produce (highly) scalable solutions



- Various examples from our recent work addressing (parts!) of the problem:
 - AURIN, ITS - data sourcing, integration & visualisation
 - Mobile Trauma Tool – eHealth
 - CONVERt – by-example data transformation & visualisation generation
 - Blue box – domain expert modelling support
 - Horus HPC – GPU-ising applications to scale
 - Yellow box – scalable compute platform
 - MDSE@R – run-time security modelling and enforcement

Examples of “social enterprise” applications

Swinburne

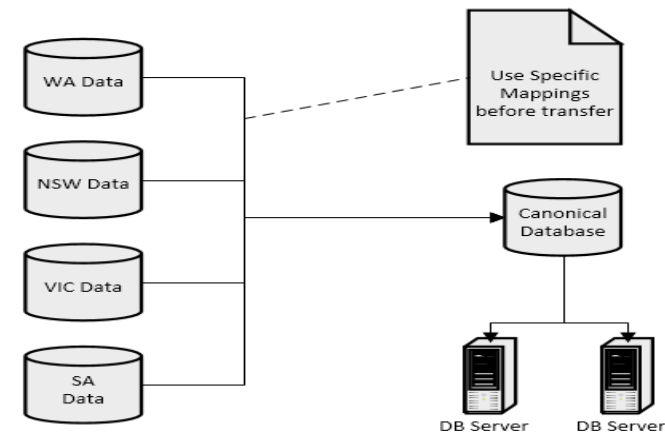


Example #1 - Information Aggregation & Visualisation

Swinburne



- Household Travel Survey data -> AURIN system
- Various states with different Survey instruments and categorisation
- Harmonise into a canonical DB
- Project includes data wrangling, mapping, and visual analytics



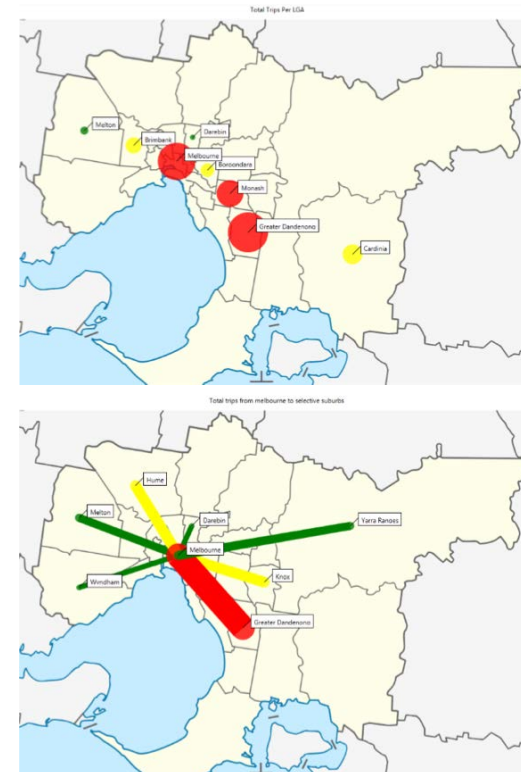
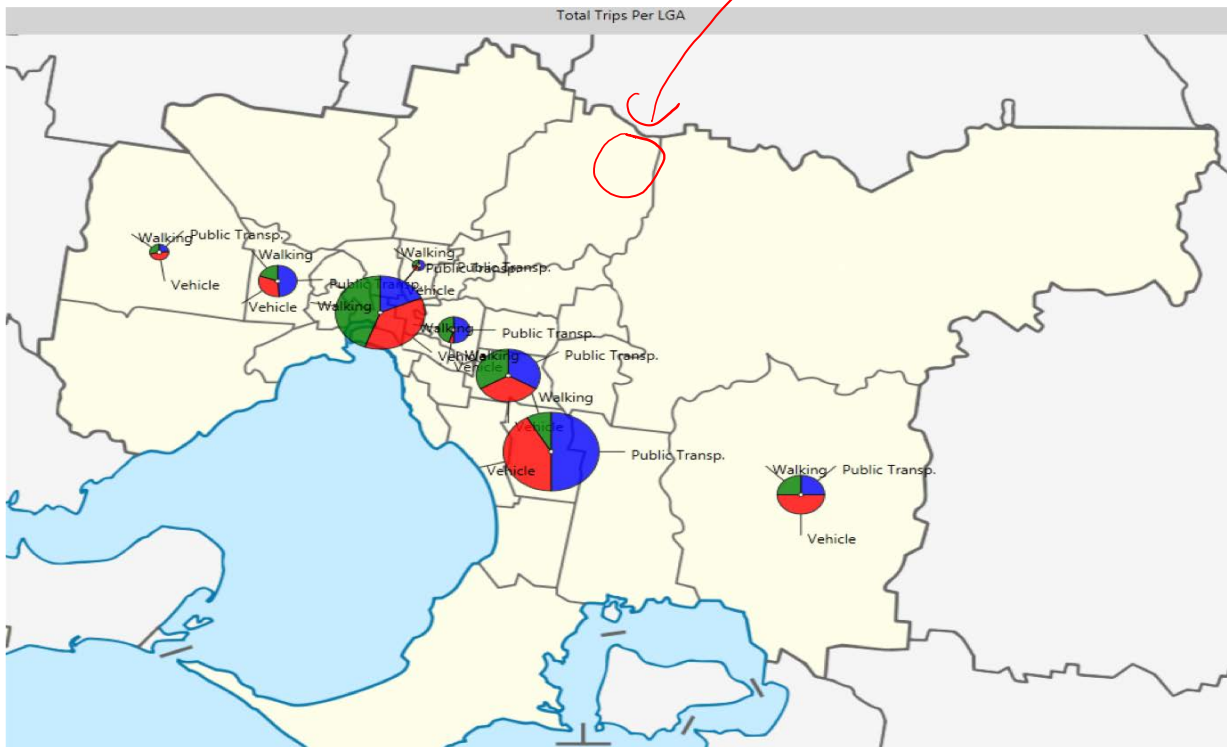
Travel purpose inconsistency samples				
VIC	NSW	Harmonised	NSW AGG	WA
N/A Missing		Missing		Default N/A Missing
Buy Something	Shopping	Buy Something	Shopping	Buy Something
Education	Education Childcare	Education	Education_ChildCare	Education Childcare
Work Related	Work related business Return to other job Return to main job Go to other job Go to main job	Work Related	Work Related Business	Work Purposes
Social	Social welfare Social visits	Social	Social_Recreation	Social
Recreational	Recreation	Recreational		Recreational

Sample Visualisations of Harmonised Data

Swinburne



JG 😊

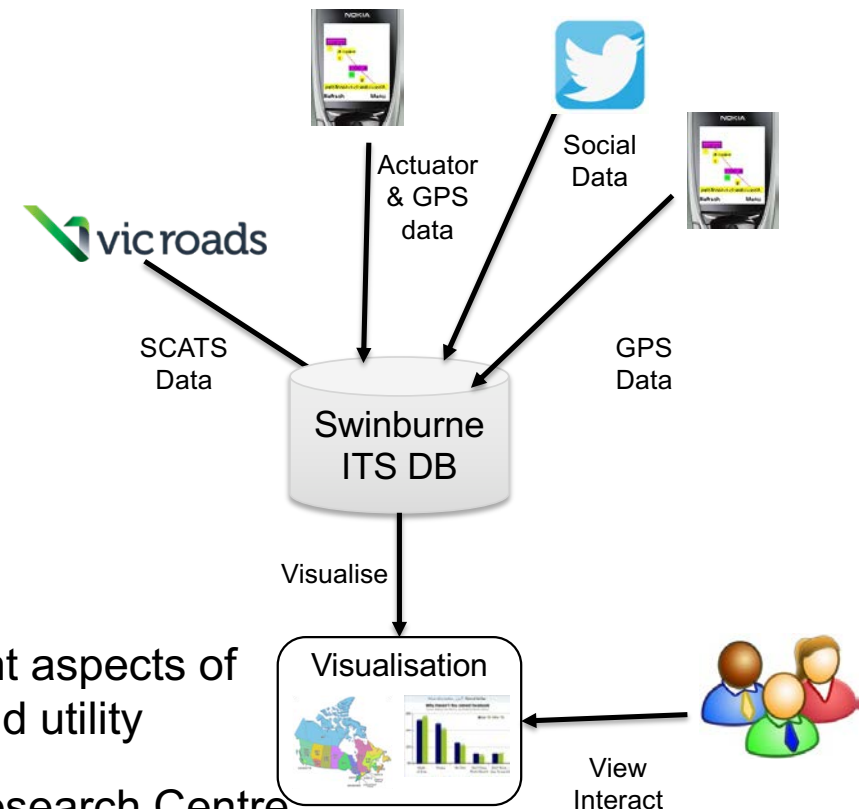


Example #2: Big Data Visualisation for Traffic Management

Swinburne

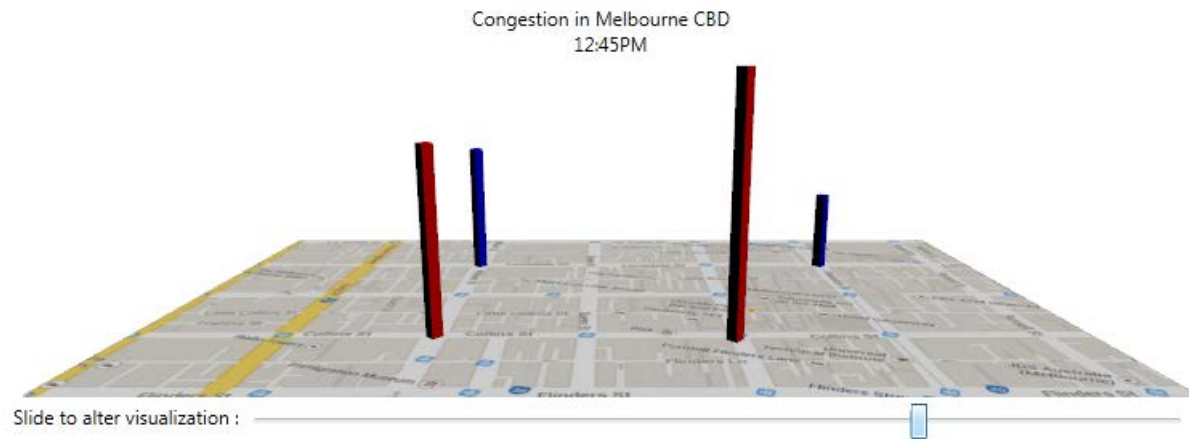


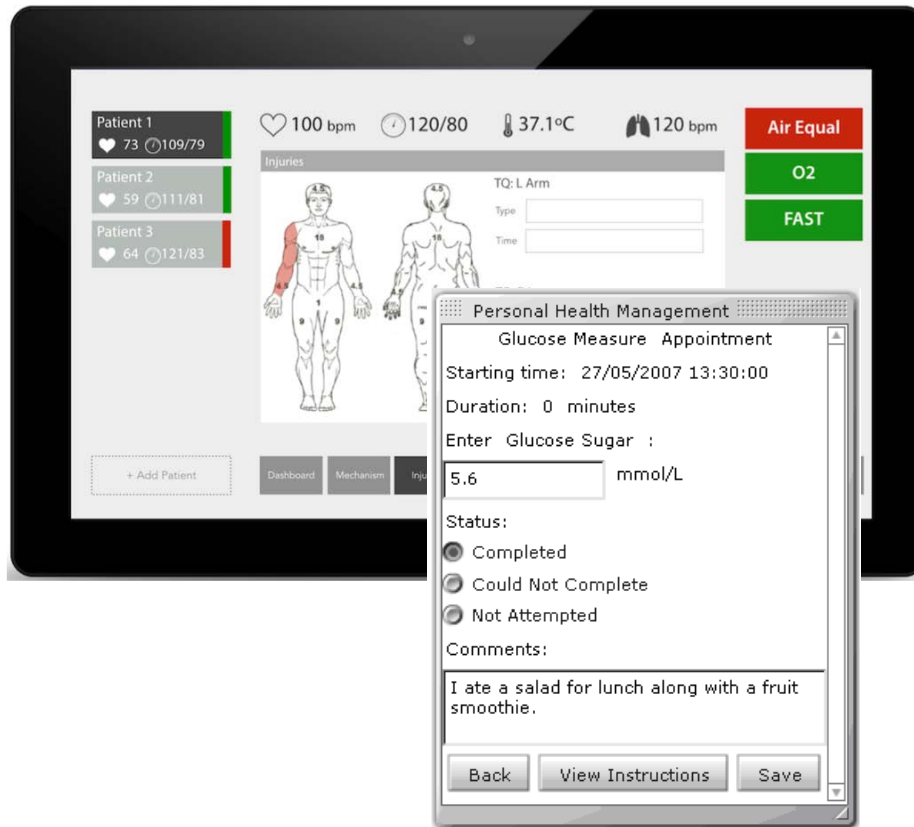
- Vicroads Collects traffic data (SCATS) - Enterprise
- Traveller GPS collects travel data – personal
- Twitter data - social
- Data is transferred to Swinburne's local DB daily
- Project:
 - Collect & integrate data
 - Visualisation of collected data
 - Users can interact with visualisations to see different aspects of the data and help traffic management – personal and utility
- Extension: PAVVET w NDSU & Great Plains Travel Research Centre – capture road quality data & vis



Sample Visualisations Traffic Data

Swinburne





- Mobile trauma tool (w Alfred Hospitals and US Navy)
- Data from various medical devices
- Visualise & decide
- Now – towards Personal health management:
 - Personal data monitoring – exercise, diet, various body and environment measures
 - How use to influence behaviour?
 - How learn to improve treatments etc??

How support specification, generation?

Swinburne



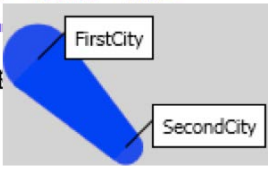
Example #1 - CONVERt- By-example data mapping & InfoVis Swinburne



```

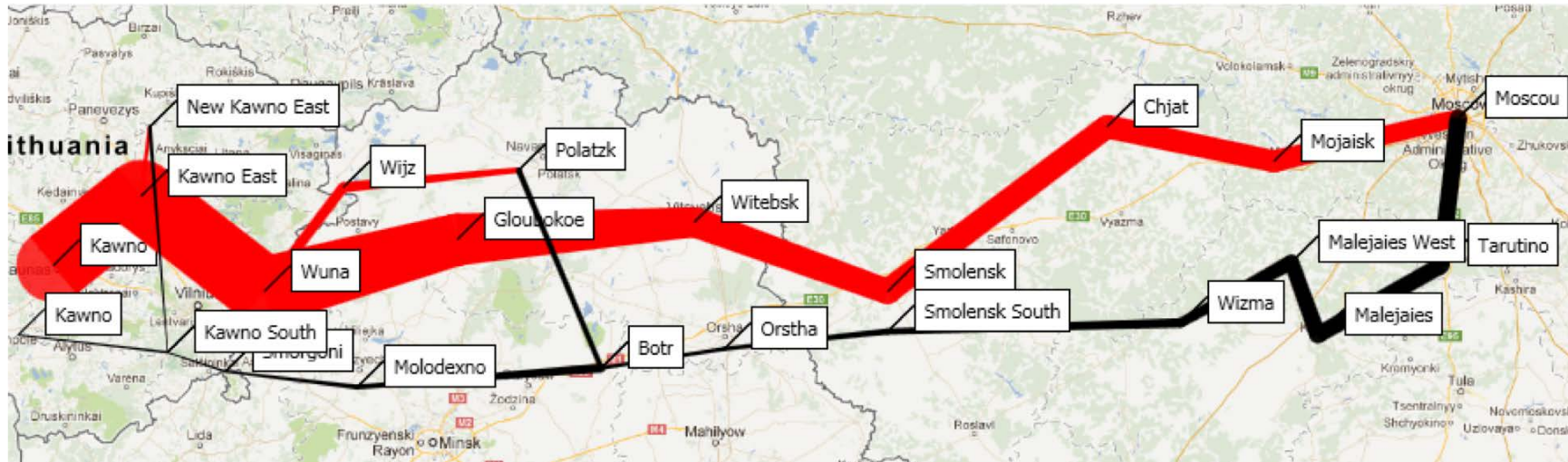
<MapData>
<Description>Discription goes here</Description>
<Movements>Troop Movements</Movements>
</MapData>
<StackPanel Orientation="Vertical" Height="290" Width="716">
<TextBlock Height="27" TextAlignment="Center"
<TextBlock.Text linkto="Description">Descrip
</TextBlock.Text>
</TextBlock>

```



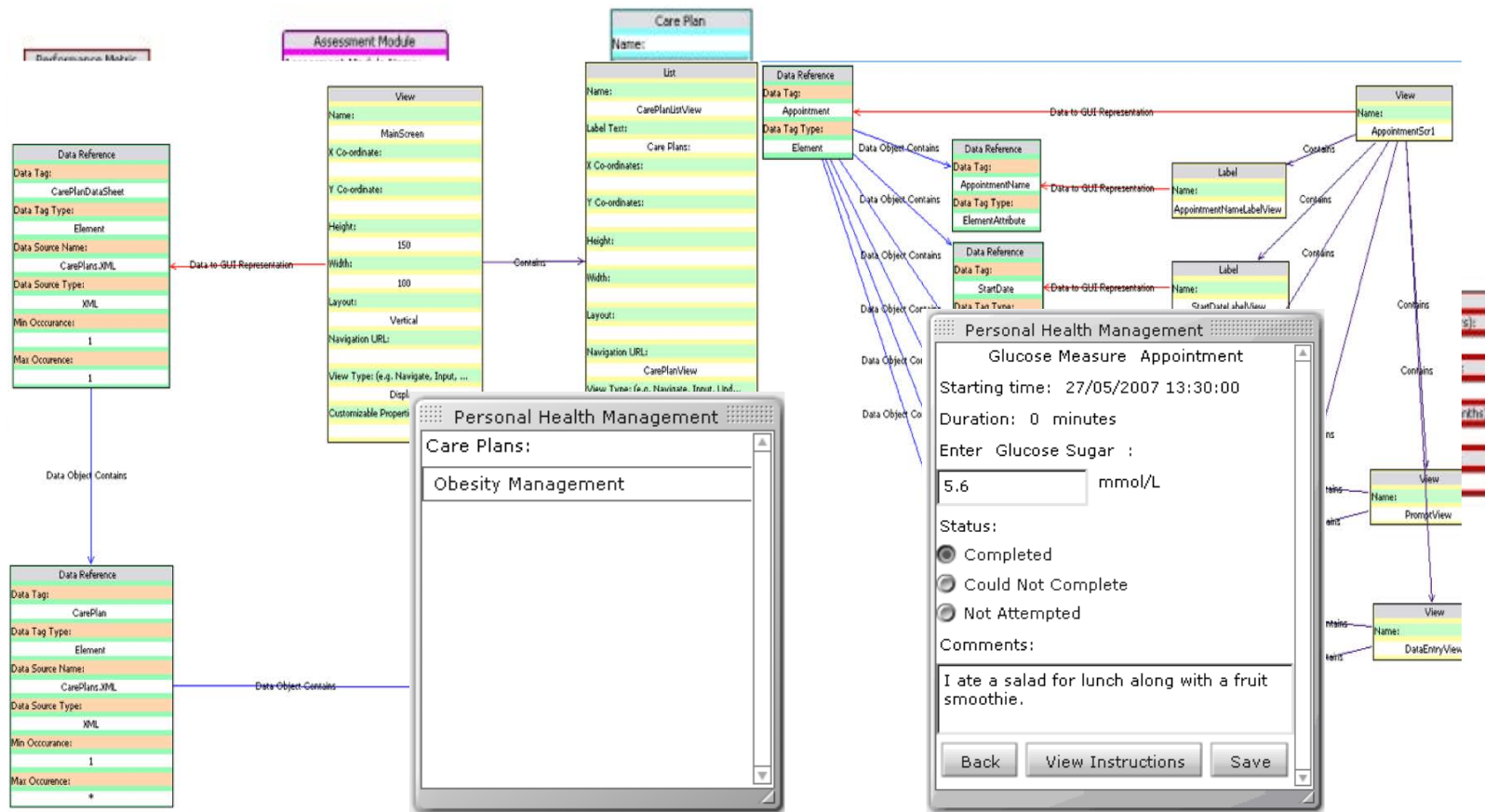
- CampaignData
 - Description
 - Figurative Map of successive losses
 - Record
 - Head
 - Start
 - Lost
 - 0

Figurative Map of successive losses in men of the French army in Russian Campaign 1812 ~ 1813



Example #2 – Visual Care Plan Modelling Language

Swinburne

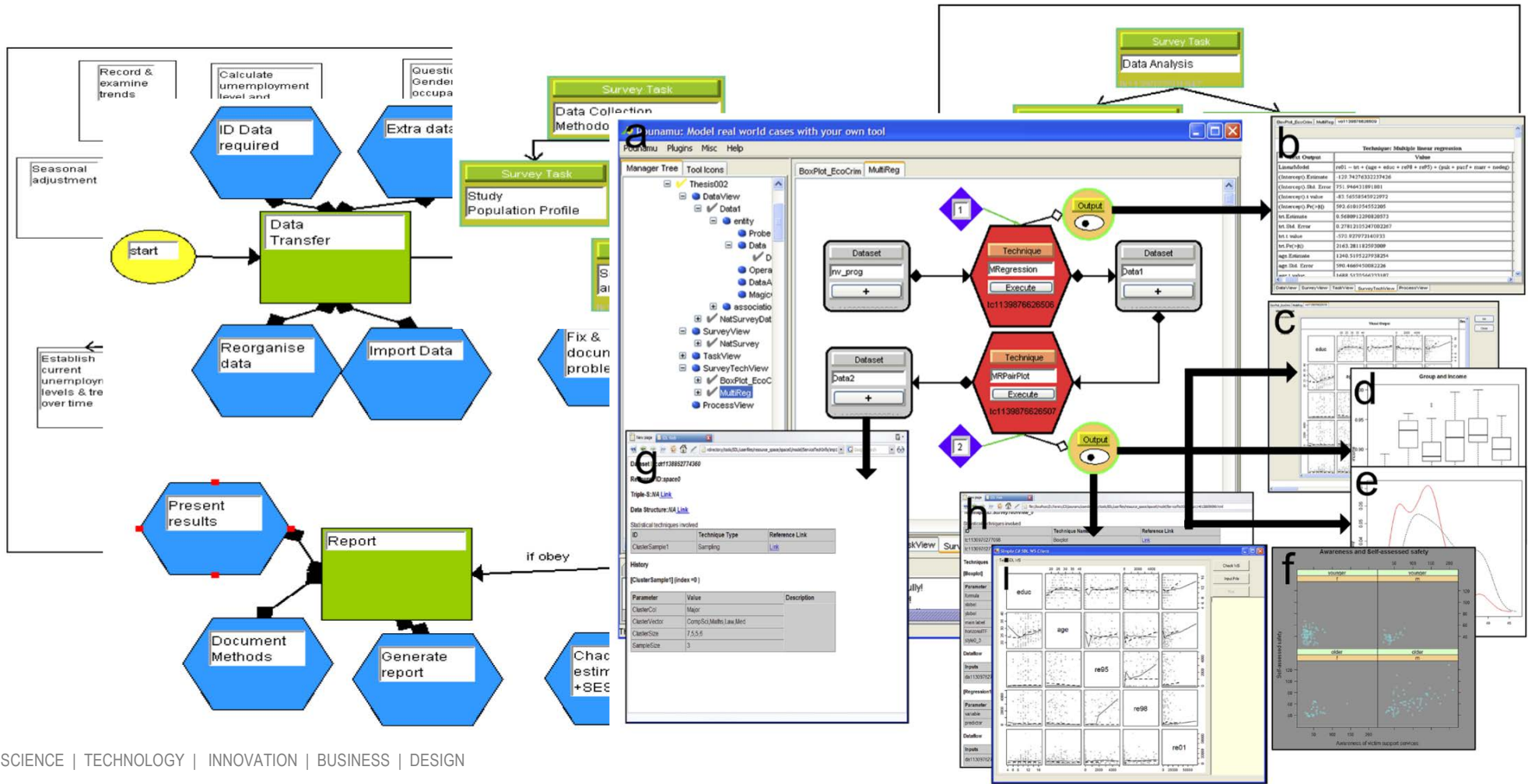




- Specifying data analytics solutions for subject matter experts
- Set of DSVLs to model aspects of complex domains:
 - Workflow in target domain
 - Disparate data sources and data formats
 - Integrated data sets
 - Re-usable data analytics abstractions expressed in domain expert concepts
 - Data visualisation specifications for decisioning
 - Generates solution to run on yellow box (or other) compute platform

Early example – Statistics Design Language Tool (SDLTool)

Swinburne

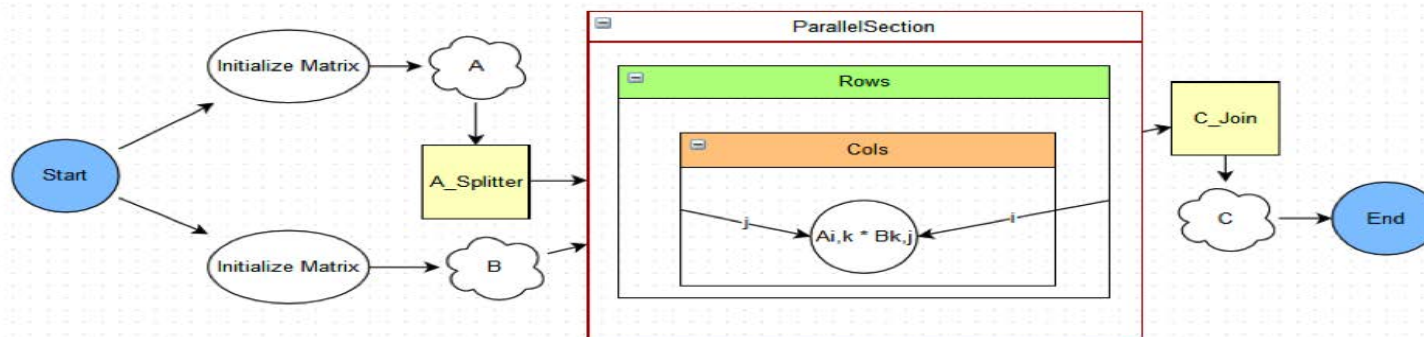
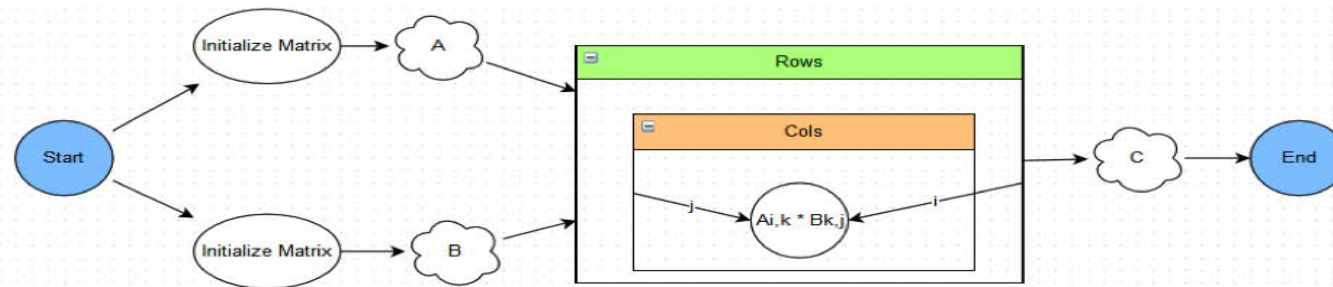




- Approach and tool for re-engineering HPC applications
- Set of DSLs to model
 - Data
 - High-level domain specific metaphors (including formulae)
 - Sequential algorithms
 - Successively specify more detailed parallel computation solutions, down to C and GPU kernel code
 - Code generators and reverse-engineering tools
 - Deployment model to run code on Grid, GPU, Cloud
 - Web-based IDE, cloud-based generation and deployment



$$C = A \times B, C_{i,j} = \text{Sum}(A_{i,k} * B_{k,j})$$





The screenshot displays a complex web-based development tool interface. On the left, a sidebar contains several tool categories: DSVDDesigner (with icons for Concept, Operation, Loop, Constant, New from, and I/O), Molecular Simulation (with icons for MaskConv, Lattice, and Initialization), Magnitic Reasonanse Imag (with icons for MaskConv, Lattice, and Initialization), General, and UML. The main workspace is divided into several overlapping windows:

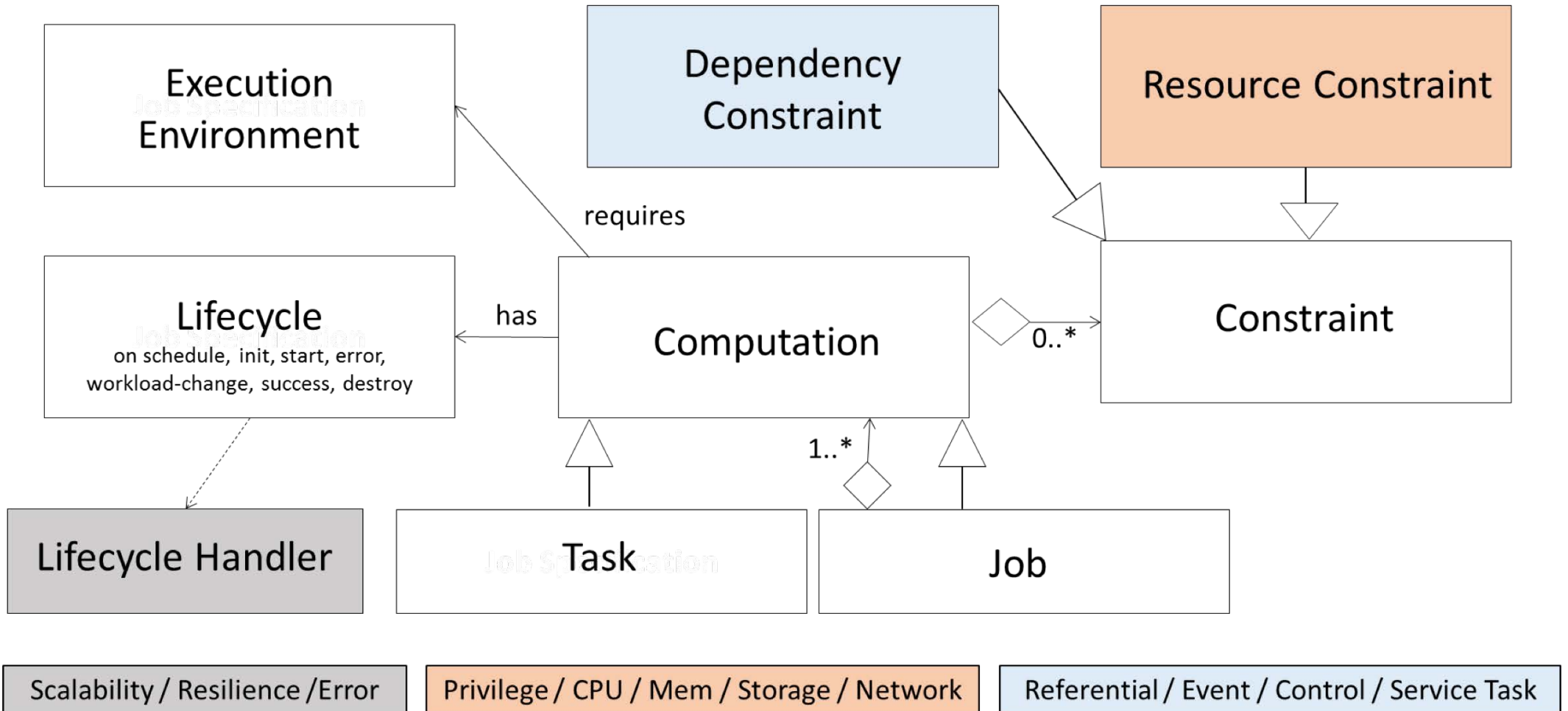
- Text Editor (top):** Contains a high-level description of the simulation process:

```
1 Part 1.2 Initial force calculations:  
2 Calculate forces (fi) on each atom from all  
3 Calculate acceleration (ai) for each atom  
4  
5 Part 2 Simulation Process:  
6 loop  
7 Part 2.1 Integrate equations of mo  
8 Part 2.1.1 Calculate fi on each atom  
9 Part 2.1.2 Apply integrator to updat  
10 Part 2.2 if (t > tEquilibration)  
11 Accumulate averages.
```
- Text Editor (bottom):** Contains C++ code for a kernel:

```
1 __kernel void sumf(__global Real4* forces,  
2 __global Real4* positions,  
3 __global struct Output* out,  
4 int count)  
5 {  
6 // Obtain force acting on id'th body  
7 const int id = get_global_id(0) + ioffset;  
8 const int thread = get_local_id(0);  
9 const int group0 = get_group_id(0);  
10 const int group = group0 + goffset; //Group 0  
11  
12 __local Real4 cache[nthreads];  
13 __local Real3 opposite[nthreads];  
14
```
- Properties:** A dialog box for configuring elements, with fields for Name, ShapeType (HTML), XMLCode, HTMLCode, Path, Addinputs, AddOutputs, and AddCommand.
- Simulation Loop:** A flowchart showing the simulation process: Lattice (represented by a ball-and-stick model) and Initialization lead into a Simulation Loop. Inside the loop, an Atoms Loop processes forces (F1, F2) and calculates acceleration a(i), velocity v(i), and position r(i).
- MaskConv and Diffing:** A diagram showing the flow of data from input A (represented by MRI scans) through a MaskConv operation to produce B. B is then used in a Diffing operation (A - B) to produce A, which is compared against a threshold. If A < Threshold, the process ends; otherwise, it loops back.

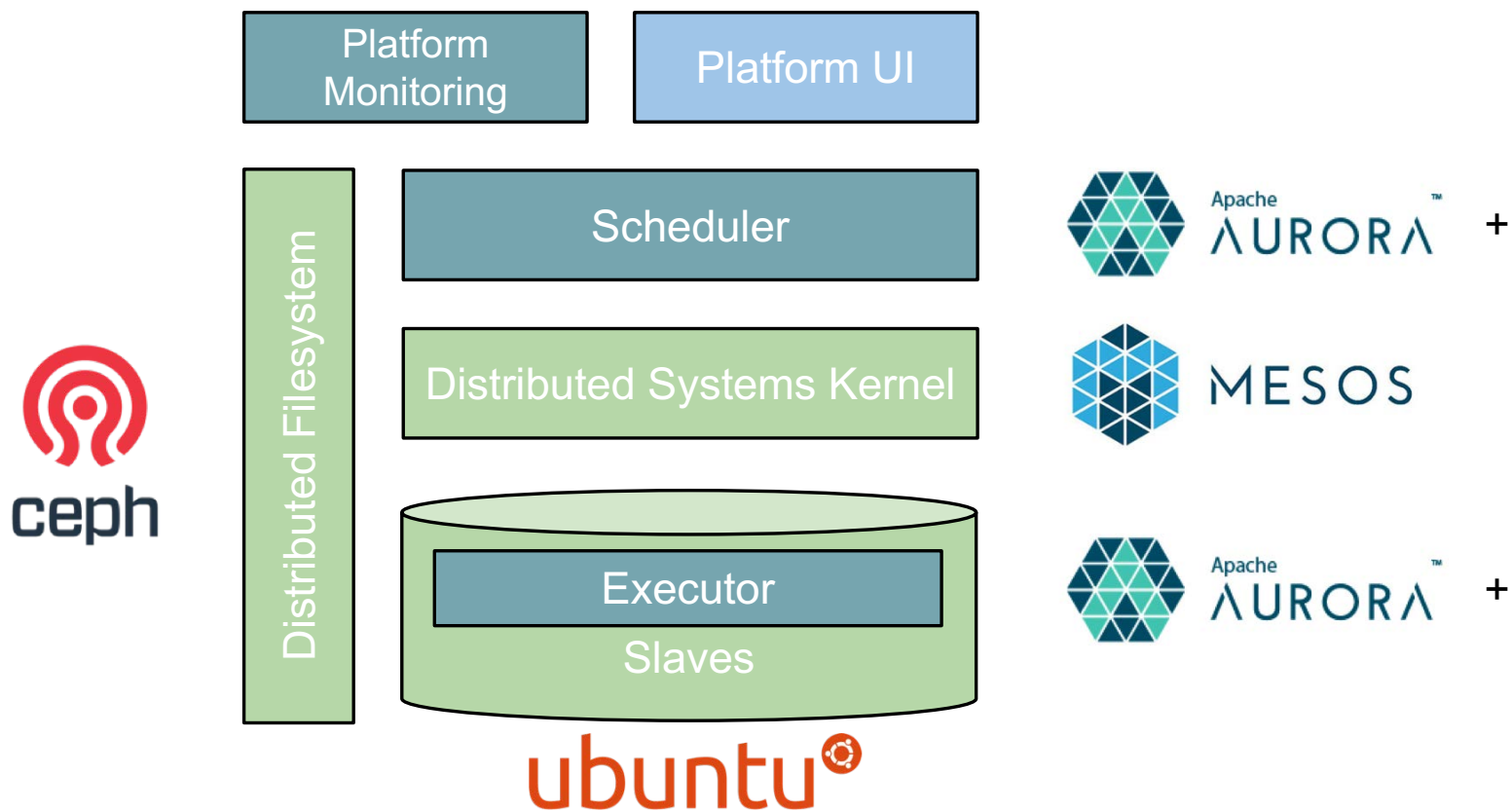


- Scalable compute platform for large data analytics problem domains
- With NICTA, DSTO, Austin Hospitals, Hawthorn Football Club, Mailguard
- Specify in set of DSVLs:
 - Deployment platform capabilities
 - Data acquisition, processing, usage workflow
 - Set of compute jobs
 - “Black box” compute components i.e. reusable parts of solution space
 - Data integration



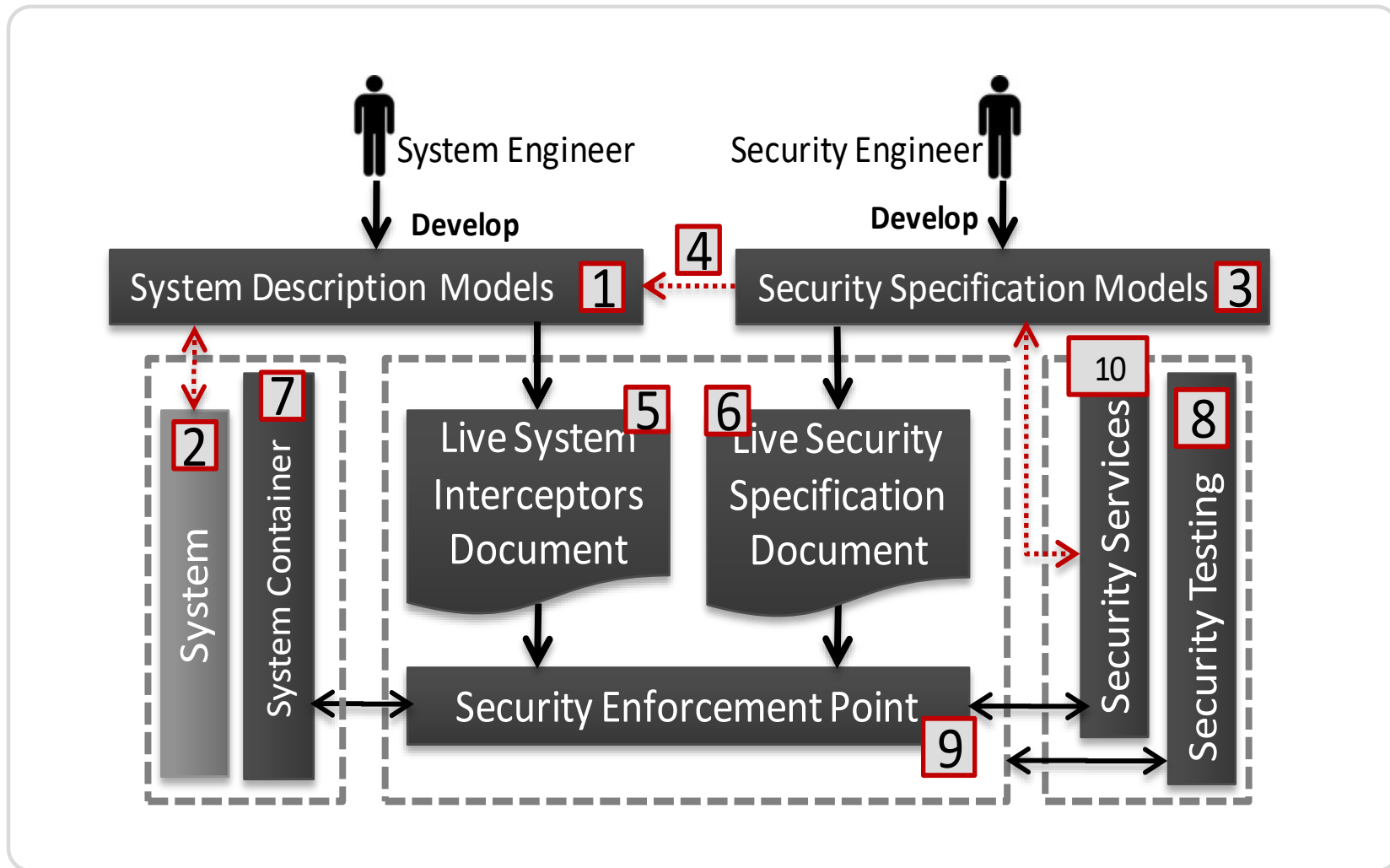
Yellow box Platform Architecture (current)

Swinburne



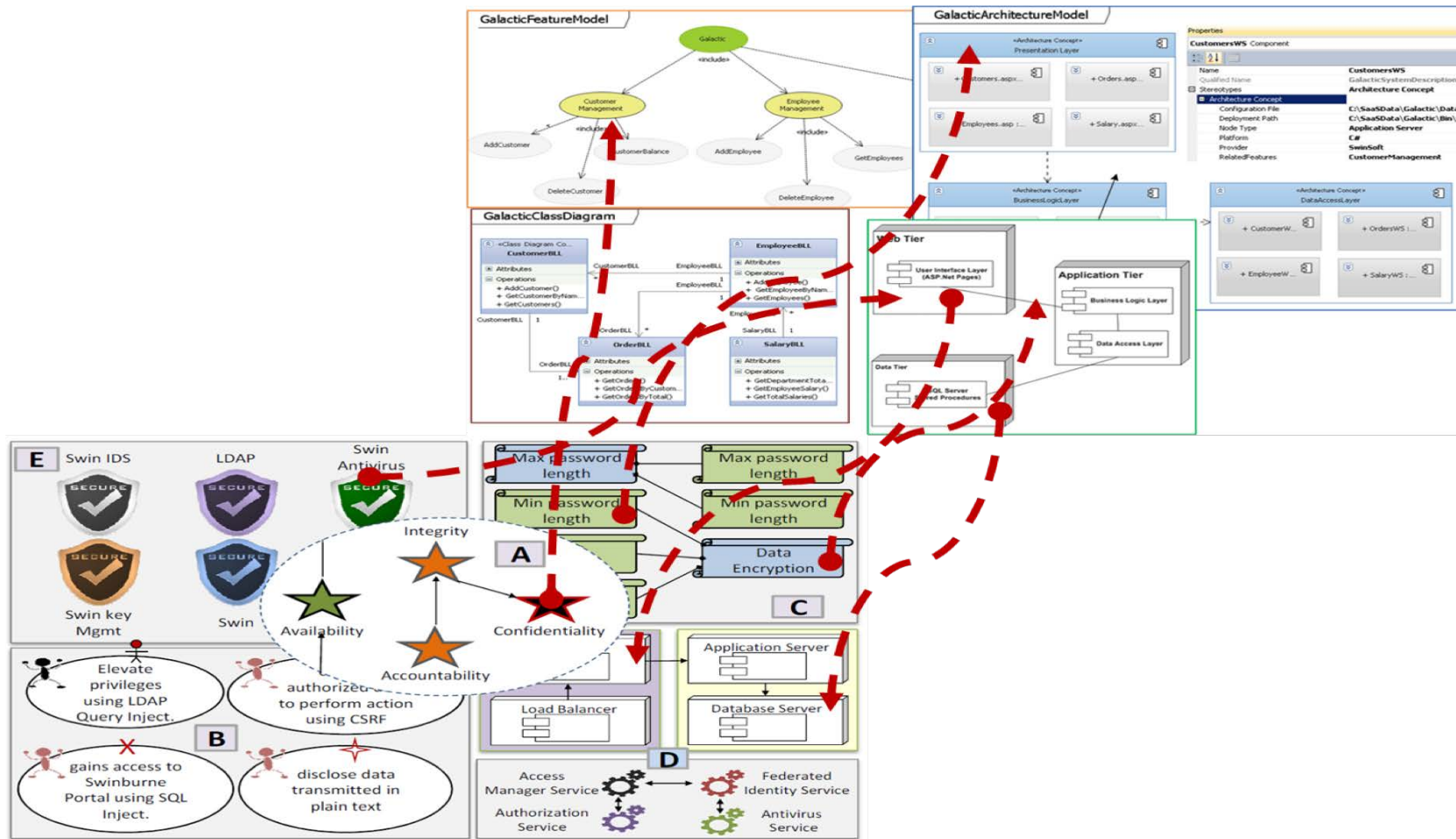
Example #6 - MDSE@R - Modelling and Generating Security

Swinburne



DSVLs supporting modelling & generation

Swinburne





- How do we model and integrate complex, heterogeneous information sources (CONVERt generalisation)
- How do we enable domain experts to specify and generate their evolving social enterprise solutions (Blue box)
- How do we generate highly scalable computations (Horus HPC, Yellow box)
- How do we scale data analytics required, especially black-box solutions (yellow box)
- How do we specify and generate effective information visualisations (COVERt, Horus HPC,)
- How do we maintain security and privacy (MDSE@R) for multi-tenant solutions with emergent users, requirements and deployment environments



- Domain expert modelling using domain-specific metaphors – Blue box, SDL++
- Diverse information sourcing and integration – purple box (sensor networks, IoT - Smart Home), CONVERt++
- Scaling of black-box solutions (yellow box) and bespoke solutions (Horus HPC)
- Better understanding of what “privacy” means esp in IoT world
- Adaptive security, especially for highly dynamic environments
- Where does personal end, social begin / social end, enterprise begin?



- The social enterprise brings new challenges and opportunities
- Access vast quantity and quality of information
- Learn behaviours, influencers, warning signs, counter-examples like could never do before
- Personal / social / enterprise / government / society lines blurred
- Security of information
- Privacy
- Just because you can build it – does it mean you should ?

Questions?

NICTA/Swinburne Software Innovation
Lab www.ssil.com.au



Ministry of Business,
Innovation & Employment



Australian Government
Australian Research Council



FRST NERF DSTools
FRST SER SPPI

DP120102653
DP140102185
LP130100201
LP130100324



Swinburne
▶ think forward



- Avazpour, I., Grundy, J.C., Grunske, L. Specifying Model Transformations by Direct Manipulation using Concrete Visual Notations and Interactive Recommendations, *Journal of Visual Languages and Computing*, Volume 28, June 2015, Elsevier, pp 195–211.
- Avazpour, I., Grundy, J.C., Vu, H. Generating Reusable Visual Notations using Model Transformation, *International Journal of Software Engineering and Knowledge Engineering*, Vol. 25, No. 2 (2015) pp. 1–29.
- Li, L, Grundy, J.C., Hosking, J.G. A visual language and environment for enterprise system modelling and automation, *Journal of Visual Languages and Computing*, vol. 25, no. 4, April 2014, Elsevier, pp. 253-277
- Kim, C.H., Grundy, J.C., Hosking, J.G. A suite of visual languages for model-driven development of statistical surveys and services, *Journal of Visual Languages and Computing*, Elsevier, Vol 26, Feb 2015, pp 99–125.
- Almorsy, M. and Grundy, J.C. Supporting Scientists in Re-engineering Sequential Programs to Parallel Using Model-driven Engineering, 1st ICSE Workshop on Software Engineering for High Performance Computing in Science (SE4HPCS 2015), Florence, Italy, May 19 2015.
- Almorsy, M., Grundy, J.C., Ibrahim, A., Adaptable, Model-driven Security Engineering for SaaS Cloud-based Applications, *Automated Software Engineering*, vol. 21, no. 2, April 2014, Springer.
- Almorsy, M. and Grundy, J.C. SecDSVL: A Domain-Specific Visual Language To Support Enterprise Security Modelling, 2014 Australasian Conference on Software Engineering (ASWEC 2014), Sydney, Australia, April 2014, IEEE CS Press.
- Khambati, A., Warren, J., Grundy, J., and Hosking, J. A Model Driven Approach to Care Planning Systems for Consumer Engagement in Chronic Disease Management, *Electronic Journal of Health Informatics*, vol 4, no. 1, 2009.