

What to expect of Business Technology Applications

- A rapid development environment that does not require all detail to be finalised
- Think people and business logic no need to think data and technology
- A unified tool to build from business speak
- No requirement to send code work offshore
- A graphical view of their actual run time business application
- The ability to change and add processes and tasks with agile software supporting constant change
- Future proof investment

Why Business Technology "BT"?

The Software Problem

- *Industry Analysts*
"Over 50% of software delivery projects fail"
- *Liam Maxwell, Centre for Policy Studies*
"The UK Government will spend about £16.5 billion on IT in 2009/10, equivalent to 1.4% of GDP. Most of this money is wasted, with only 30% of projects succeeding"
- *Standish Chaos 2009 report on IT projects*
"This year's results show a marked decrease in project success rates, with 32% of all projects succeeding which are delivered on time, on budget..."
- Worldwide cost of IT failures is \$6.2trillion according to Roger Sessions.

21st Century Thinking

- *George Colony, CEO Forrester Research*
"If we don't get from IT to BT we're going to have more disasters like our present mortgage meltdown. Why? Because **IT creates impenetrable systems that human beings can't manage. BT is about human beings back in control.**"
- *Pekka Huttunen, Senior Executive with Accenture High –Tech Group, September 2009*
"Emphasis should be on **an integrated technology stack that supports vertical applications and standard and custom offerings**"

In technology terms what is required?

- Recognising there are relatively **few work task types including the user interface** (human and system) that can address **any business issue** creating a function independent capability handling all business logic **where the core code never changes**
- **Separation of business logic** to allow businesses to take control of their business with IT in support to deploy
- Data centric architecture delivers real process **agility in the software** to change as the business changes with no disruption
- Building **direct from the model** at a click of a button once the business logic is tested with no code generation or compiling using a declarative technique
- Build very rapidly a Prototype led by the analysts from **simple business language** and **not requiring a traditional finalised technical specification**
- The model becomes a **permanent record** of what is deployed
- A unified technology stack that **removes complexity** to build custom applications

A unified tool required to deliver future proof Applications

- Process engine *to ensure all works to plan*
- Rules engine *reflecting real world of compliance*
- Calculation engine *automating system work*
- State engine *Real time feed back*
- BPM *focus on people and their processes*
- Workflow *everything connected in right order*
- Audit trail, events, escalations = control
- Rapid prototyping *user involvement in build*
- Time recording *supports activity based costing*
- Real time reporting *become predictive*
- Build mash ups *one screen multiple data sources*
- Linked Ajax Grids *faster access to related data*
- Roles and performers *people and machines*
- Management hierarchy *who sees what*
- Orchestrating business processes in SOA box *a 21st century approach = agility in software*
- Call Web Services *wrapped up in a process*
- Reusable features *speeds up future development*
- User interface dynamically created linking people, roles, task type and data via forms using web or client server
- Presentation layer with TAG library
- Pre-built templates for custom documents, letters, e-mails, messages etc dynamically populated with instance specific data and edit capability in browser
- Simplified table creation in Oracle Database
- Process and task versioning control
- Business analysts/consultants build business logic with business
- Server Side Message Queue
- XML runtime and design Gateways
- SaaS, Platform as-a-service and Cloud Computing all supported

Look for delivery on what Bill Gates sees at the future of business software

Microsoft's 2008 Office System Developers Conference in San Jose, Calif.

“Most code that’s written today is procedural code. And there’s been **this holy grail of development forever, which is that you shouldn’t have to write so much [procedural] code,**” Gates said. “We’re investing very heavily to say that customization of applications, **the dream, the quest,** we call it, should take **a tenth as much code as it takes today.**”
“You should be able to do things on a declarative basis,”

“We’re not here yet saying that [a declarative language has] happened and you should write a ton less procedural code, **but that's the direction the industry is going,**” Gates said. “And, despite the fact that it’s taken longer than people expected, we really believe in it. **It’s something that will change software development**”