IBM

IBM Software

Innovate2011

The Premier Event for Software and Systems Innovation

Software. Everyware.

19- July, Sydney, Australia
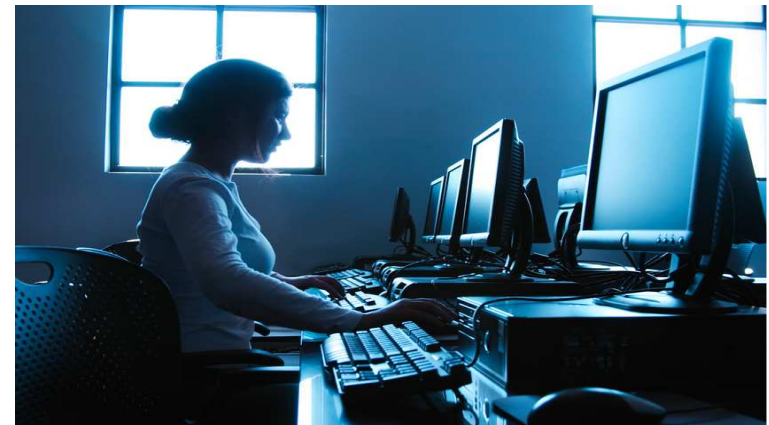**21- July,** Melbourne, Australia

- The need for WebSphere Automation
- Overview of Build Forge and RAFW
- Demonstration
  - Building a Build Forge Project
  - WebSphere Installation
  - WebSphere Configuration
  - WebSphere Deployment
- Summary

Key Challenges Around Managing WebSphere Deployments and Environments

- Existing admin/deployment teams unable to scale with expanding application portfolios

- Largely manual, time consuming and labor intensive administration and deployments

- Configuration errors and defects that are frequent and difficult to both find and fix

- Increasing compliance mandates and manual compliance processes

- Lack of a disaster recovery option / plan

Factors that Effect WebSphere Deployments and Admin Today

24x7

Best Practices for WebSphere Administration

- There are many Best Practice procedures recommended by WebSphere but not implemented within the product
  - Backup current version of application before updating with new build
  - Use source control to manage & control WebSphere configuration settings
  - Embed build id in J2EE EAR file Manifest.mf file as it is built
  - Fall back to previous version of the application if the update fails or is unacceptable
  - Store a record of who updates the application and when (and why)

- Combination of Build Forge and Rational Automation Framework for WebSphere can automate these Best Practices to ensure that they are guaranteed to be executed

IBM Rational Automation Framework

Build Forge
(Rational Automation Framework)

+

Rational Automation Framework for WebSphere

- Automates Tasks
- Scheduling
- Notifications
- Auditability
- Visibility
- Consolidation & Standardization
- Role-based Security

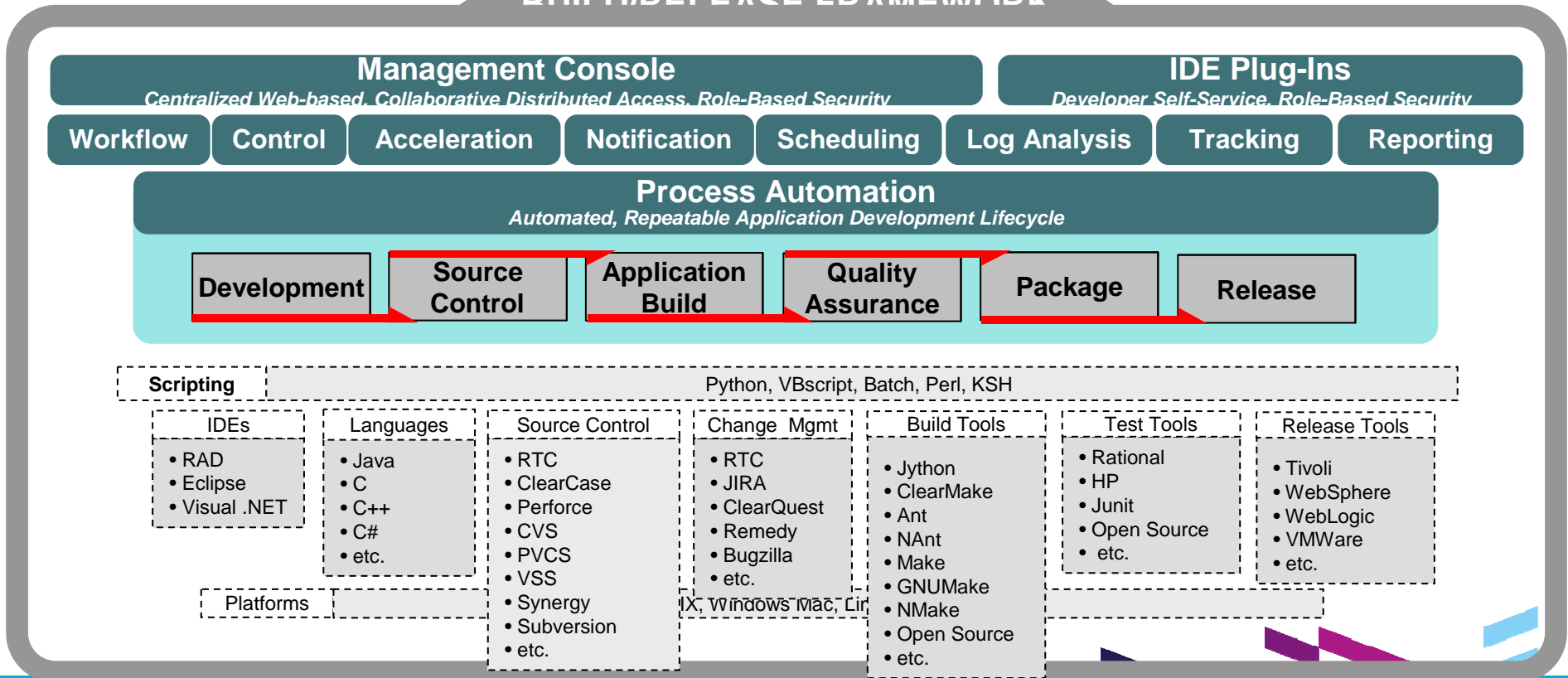- Library of 900+ commonly used tasks to administer WAS, Portal, WVE, Process Server, ESB, and WSRR
- Environment Generation Wizard
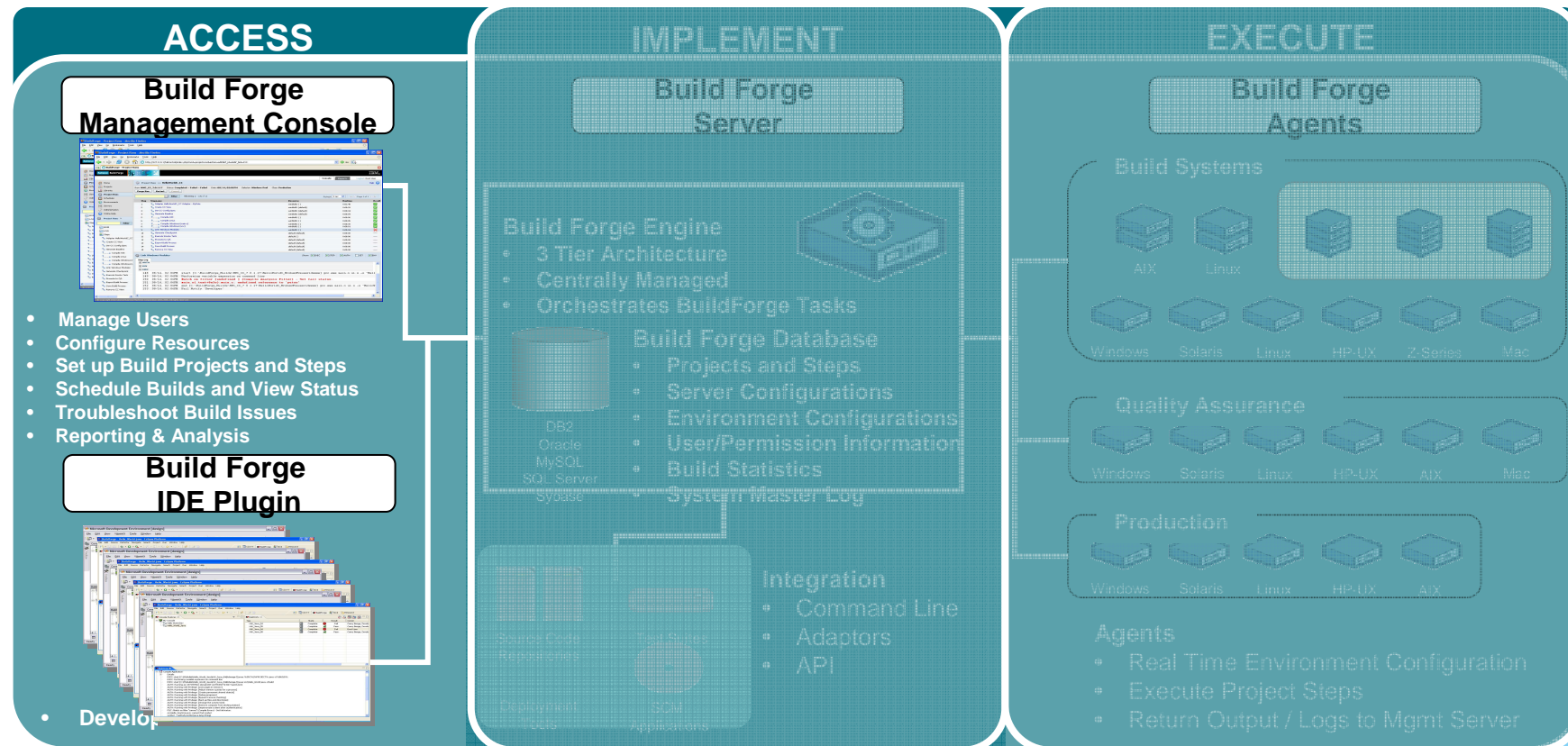- Predefined Automation Library elements

IBM Rational Build Forge

BUILD/RELEASE FRAMEWORK

Management Console
Centralized Web-based, Collaborative Distributed Access, Role-Based Security

IDE Plug-Ins
Developer Self-Service, Role-Based Security

Workflow	Control	Acceleration	Notification	Scheduling	Log Analysis	Tracking	Reporting

Process Automation
Automated, Repeatable Application Development Lifecycle

Development	Source Control	Application Build	Quality Assurance	Package	Release

Scripting — Python, VBscript, Batch, Perl, KSH

IDEs	Languages	Source Control	Change Mgmt	Build Tools	Test Tools	Release Tools
• RAD	• Java	• RTC	• RTC	• Jython	• Rational	• Tivoli
• Eclipse	• C	• ClearCase	• JIRA	• ClearMake	• HP	• WebSphere
• Visual .NET	• C++	• Perforce	• ClearQuest	• Ant	• Junit	• WebLogic
	• C#	• CVS	• Remedy	• NAnt	• Open Source	• VMWare
	• etc.	• PVCS	• Bugzilla	• Make	• etc.	• etc.
		• VSS	• etc.	• GNUMake		
		• Synergy		• NMake		
		• Subversion		• Open Source		
		• etc.		• etc.		

Platforms — IX, Windows Mac, Lin

How it Works

ACCESS

Build Forge
Management Console

- Manage Users
- Configure Resources
- Set up Build Projects and Steps
- Schedule Builds and View Status
- Troubleshoot Build Issues
- Reporting & Analysis

Build Forge
IDE Plugin

- Develop

IMPLEMENT

Build Forge
Server

Build Forge Engine
- 3 Tier Architecture
- Centrally Managed
- Orchestrates BuildForge Tasks

Build Forge Database
- Projects and Steps
- Server Configurations
- Environment Configurations
- User/Permission Information
- Build Statistics
- System Master Log

DB2
Oracle
MySQL
SQL Server
Sybase

Integration
- Command Line
- Adaptors
- API

Source Code
Repositories

Deployment
SCM
Applications

EXECUTE

Build Forge
Agents

Build Systems
- AIX
- Linux
- Windows
- Solaris
- Linux
- HP-UX
- Z-Series
- Mac

Quality Assurance
- Windows
- Solaris
- Linux
- HP-UX
- AIX
- Mac

Production
- Windows
- Solaris
- Linux
- HP-UX
- AIX

Agents
- Real Time Environment Configuration
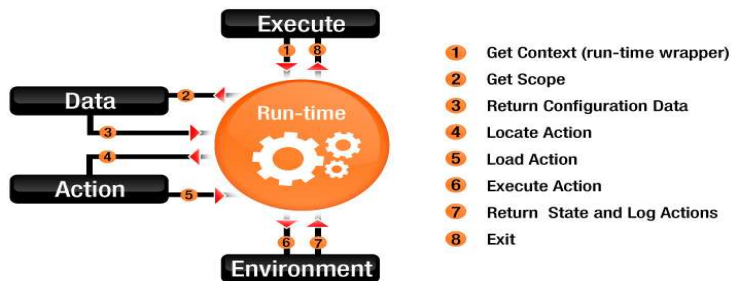- Execute Project Steps
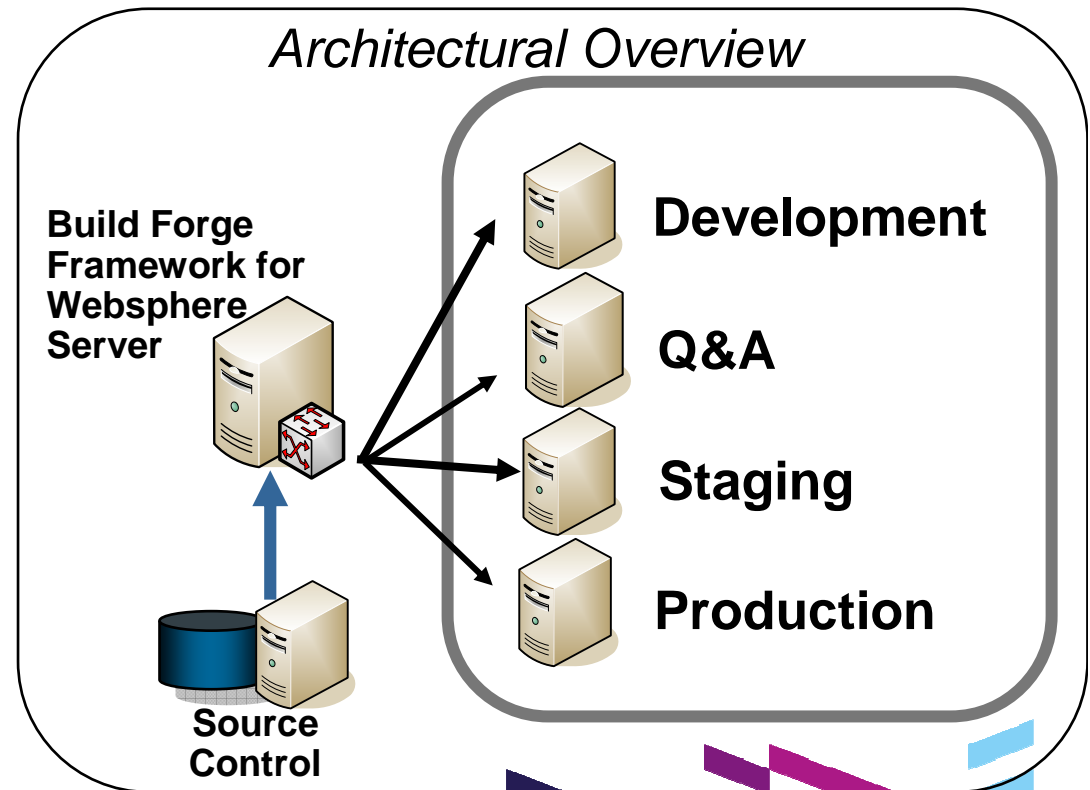- Return Output / Logs to Mgmt Server

RAFW: Automation of WebSphere & Portal Admin

- Common use scenarios
  - Multi-cell WebSphere tool
  - Environment build out
  - Configuration capture
  - Application deployment

Run-time Overview

	Get Context (run-time wrapper)
1	Get Context (run-time wrapper)
2	Get Scope
3	Return Configuration Data
4	Locate Action
5	Load Action
6	Execute Action
7	Return State and Log Actions
8	Exit

Architectural Overview

Build Forge Framework for Websphere Server

Development

Q&A

Staging

Production

Source Control

Achieving The Software Factory

- Software production as a "service"
- Standardize and reuse across teams
- Remove obstacles to distributed development

Time-to-Market Advantages

- Remove the software production bottlenecks
- Automate the assembly process
- Remove manual handoffs

Visibility, Security, and Auditing

- Avoid the dreaded failed application audit
- Get audit control and traceability
- Secure the build environment

Pushing the Quality Agenda Upstream

- Hold developers to a higher level of quality
- Identify & resolve quality issues faster
- Enable agile development

The Value of RAFW

Reduce Maintenance Time

- Minimize deployment/environment build time
- Build Environments in hours not days
- Deploy code in minutes not hours

Gaining Control of your Environment

- Eliminate Configuration drift problem
- Enforce Standards Across the Enterprise
- Put Configuration under Change Control

Repeatability & Consistency

- Rebuild environments consistently
- Reliable and Repeatable Disaster Recovery
- Eliminate Human Errors

Reduce Costs

- Free staff up for strategic initiatives
- Reduce labor costs
- If Time = Money, Less time = More Money

IBM

Questions