
Informix Special Builds

Dr.Andreas.Dworsky@de.ibm.com



Agenda

- Motivation
- Special Build – Anmerkungen
- Versionsbezeichnung – X
- Notwendige Informationen
- Erstellung – Release-Verwaltung
- Erstellung – Merge & Build
- Erstellung – Nach dem Build
- Merge – Überblick
- Merge – Datei
- Wissenswertes – Gestern und Heute
- Zusammenfassung
- Fragen

Motivation

- Existierenden Defekt-Fix schnellstmöglich einsetzen
- Eventuell existierende reguläre Version mit Defekt-Fix nicht verwendbar
 - Fehlende Third-Party Zertifizierung
 - Tests noch nicht (vollständig) durchgeführt
- Special Build basiert auf im Einsatz befindlicher Version
 - Zusätzlicher Defekt-Fix bedeutet minimale Änderung
 - ... damit auch minimales Risiko – in der Regel

Special Build – Anmerkungen

- Zu beachten ist: Ein Special Build ist keine reguläre Version
 - Keine QA Tests seitens IBM
 - Nicht kumulativ
 - Empfehlungen:
 - Intensiver Test in einer Integrations-Umgebung
 - Einsatz eines Special Builds nur bis zum Erscheinen der regulären Version mit dem entsprechenden Defekt-Fix

Versionsbezeichnung – X

- Versionsbezeichnung Beispiele
 - 11.50.FC9**X1**
 - 11.70.FC7W3**XA**
 - 12.10.FC5W1**X7WE**
- Versionsbezeichnung nicht 100% einheitlich
 - 11.50.FC9X1 könnte basieren auf
 - 11.50.FC9
 - 11.50.FC9W1
 - ...
- Nicht kumulativ
 - 12.10.FC6X2 enthält nicht (notwendigerweise) die Änderung von 12.10.FC6X1, etc.
- Nicht Plattform übergreifend
 - 12.10.FC6X1 für Linux enthält nicht (notwendigerweise) die gleichen Defekt-Fixes wie 12.10.FC6X1 für AIX
- Diagnose-Builds: **D**
 - 12.10.FC6**D1**

Notwendige Informationen

- Business-Case
- Basis-Version (oninit -V, oninit -version)
- Plattform (uname -a)
- Edition
- APAR-/Defekt-Fixes
 - Voraussetzung ist, dass die Defekt-Fixes existieren
 - ... in der jeweiligen Version
 - Falls der Defekt-Fix in der Version noch nicht vorliegt
 - ... muss von Development ein Backport erfolgen

Erstellung – Release-Verwaltung

- Existierendes Projekt im Release-System "klonen"
- ... und mit neuer Versionsbezeichnung erstellen
- Release-Informationen eintragen
 - APAR-/Defekt-Nummern
 - PMR-Nummer
 - Edition
 - ...
- "Elements" aller Defekt-Fixes zusammenstellen
- Configspec erzeugen lassen
 - ClearCase Branch

Erstellung – Merge & Build

- View Configspec setzen
- Files auschecken
- Merge der Defekt-Fixes durchführen
- Merge kontrollieren und gegebenenfalls manuell nacharbeiten
- Files einchecken
- Build starten
- Build kontrollieren

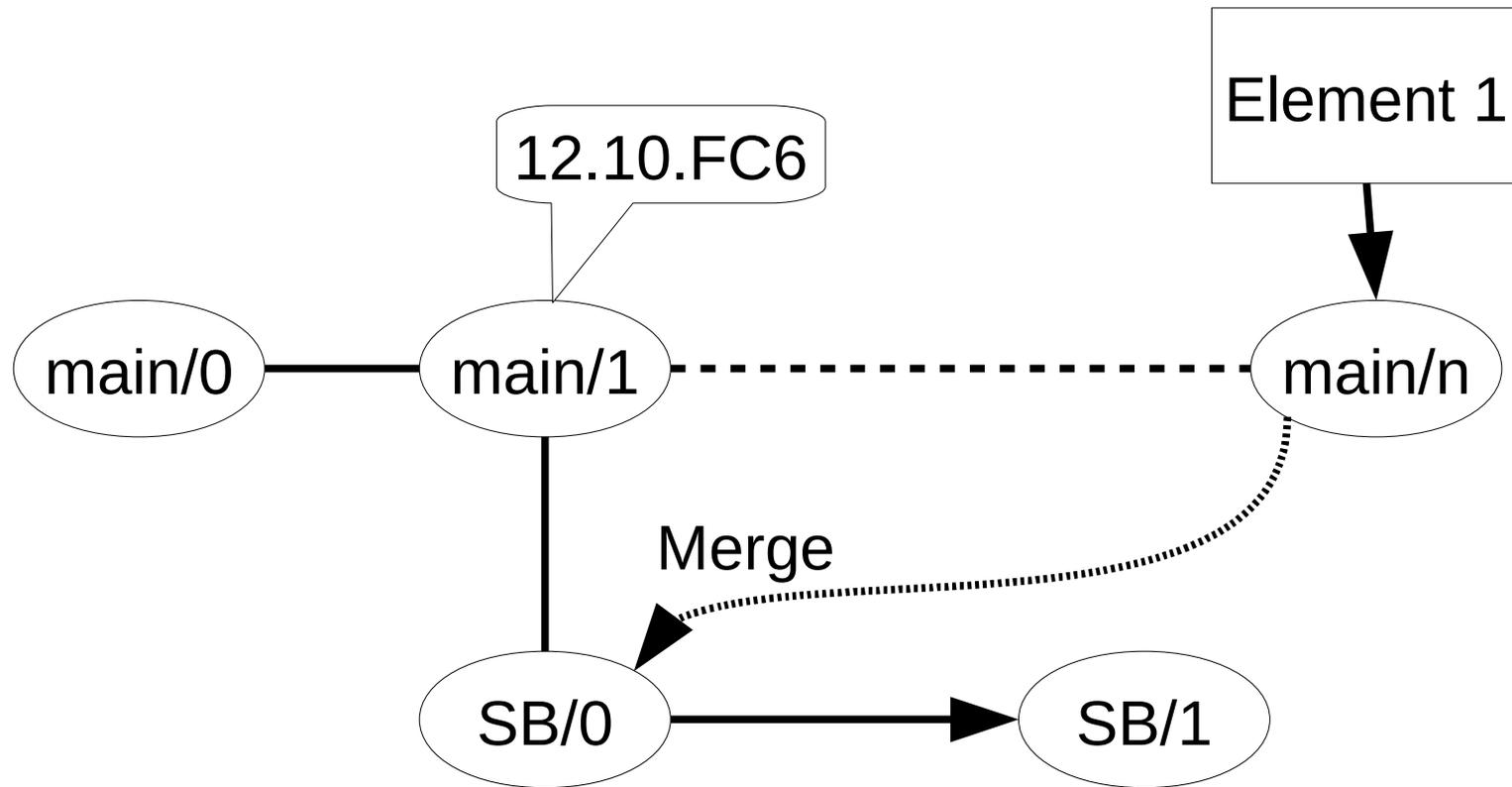
Erstellung – Nach dem Build

- Gewünschte Edition(en) bereitstellen
- Installation für den "Basic Sanity" Test
- Vorgang teilweise automatisiert

Merge – Überblick

- Merge kann erschwert werden durch
 - Anzahl der Files
 - Differenz zwischen Version für den Fix und der Basis
 - Abhängigkeiten des Defekt-Fixes von Code-Änderungen
- Berücksichtigung von “Regressions“, sofern bekannt

Merge – Datei



Wissenswertes – Gestern und Heute

- Special Build Team
 - In der Vergangenheit: Development - Platform Engineering
 - Heute: Advanced Support
- Auslieferung
 - In der Vergangenheit: Support
 - Heute: Customer Support Operations
- Produkt
 - In der Vergangenheit: Einzelne Binaries
 - Heute: Server-Install

Zusammenfassung

- Special Build kann kurzfristige Lösung bedeuten
- ... sofern keine anderen Optionen existieren
- Special Build sollte von einer reguläre Version abgelöst werden
- ... wenn die Möglichkeit besteht
- Erstellung eines Special Builds
 - Basis-Version, Plattform, Edition, APAR-/Defekt-Liste
 - Manueller oder teil-automatisierte Merge und Build
 - Minimaler Funktionstest, keine QA

Fragen?

Dr.Andreas.Dworsky@de.ibm.com

