

Enterprise & Database CM: a process-based technique

Austin Hastings
Principal Consultant
Longacre, Inc.



Introduction

- The name is LDM
 - Longacre Deployment Management
- Developed for database-centric enterprise project
- Turns out to have enterprise benefits

Database CM is *hard*

- Databases CM is about incremental changes.
- File CM is about picking versions (files get replaced entirely).
- Many databases cannot be modelled well by a directory tree.
- Most CM tools don't even handle directories well.

DBCM: Starter kit

- Create a “script” for each set of changes.
- Treat each table, proc, trigger as an object.
Deliver separate updates.
- Use a tool to “synchronize” your development and build schemas.

Tools you can use

- MySQL Workbench
- TOAD
- Oracle Developer
- dbDeploy
- Liquibase
- XML to DDL



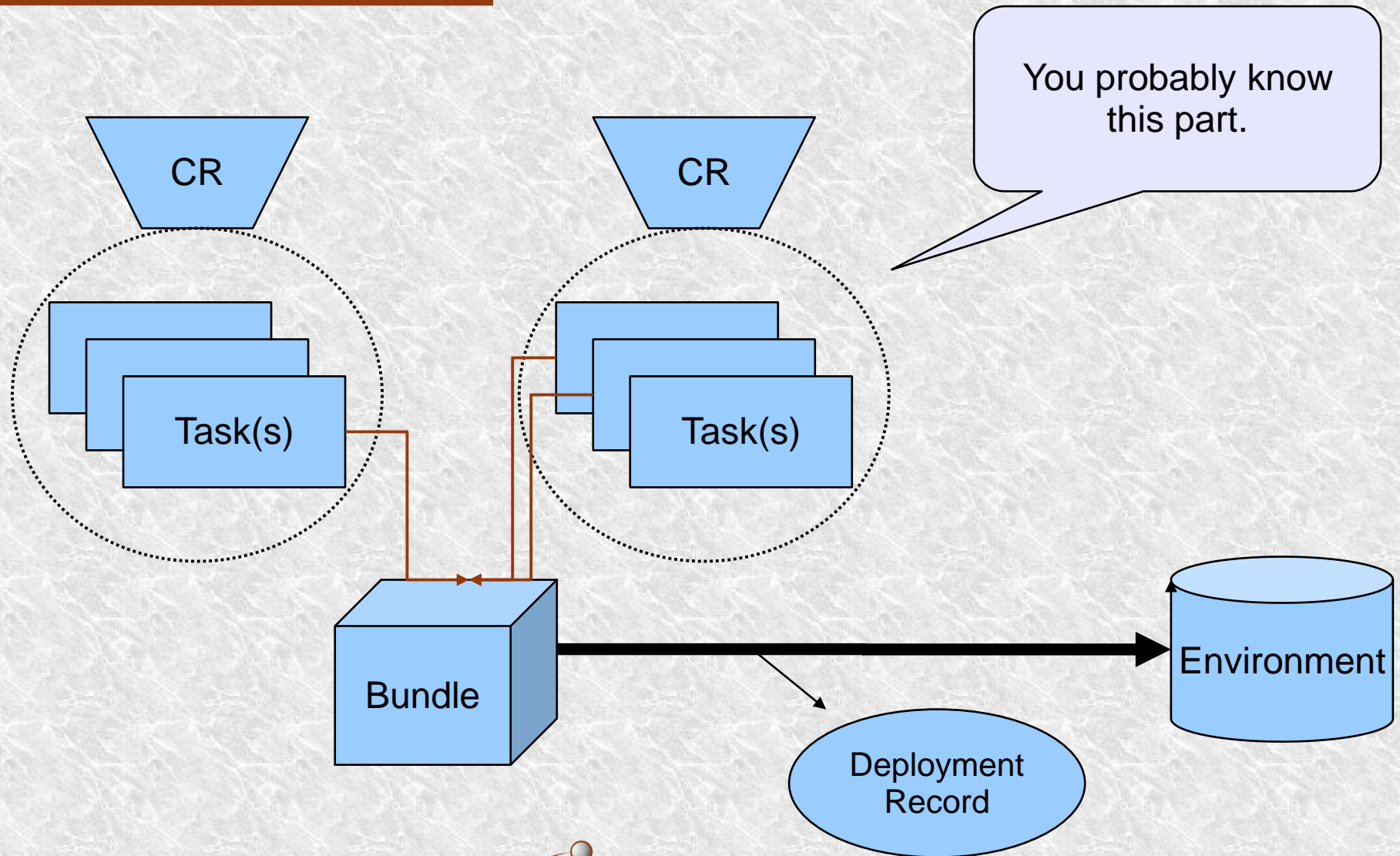
LDM: A process based approach

- Do one of the above. Any one.
- Use a *very powerful* life cycle tool to automate it.
- You need separate work flows for different “classes” of object.
- You need good reporting on relationships (links).

Tools you can use

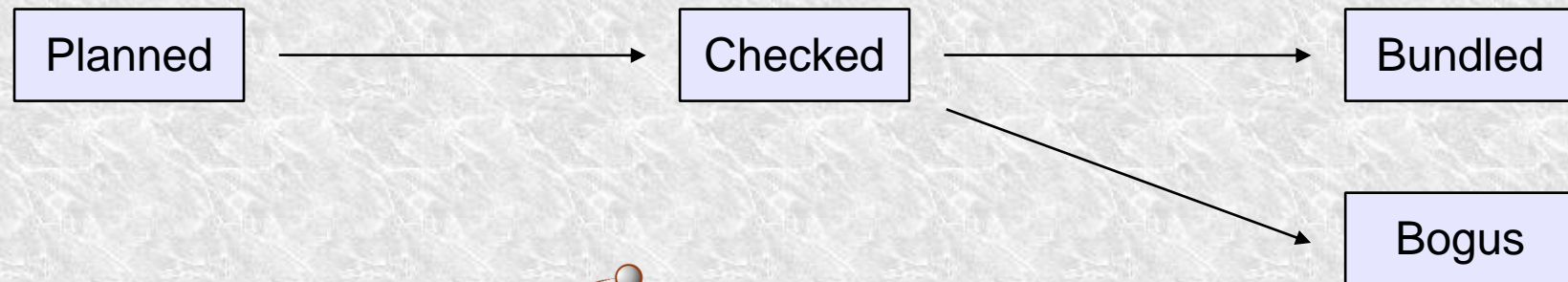
- Telelogic Change Synergy
- MKS Integrity
- Jira (not so good)

Several types of entity



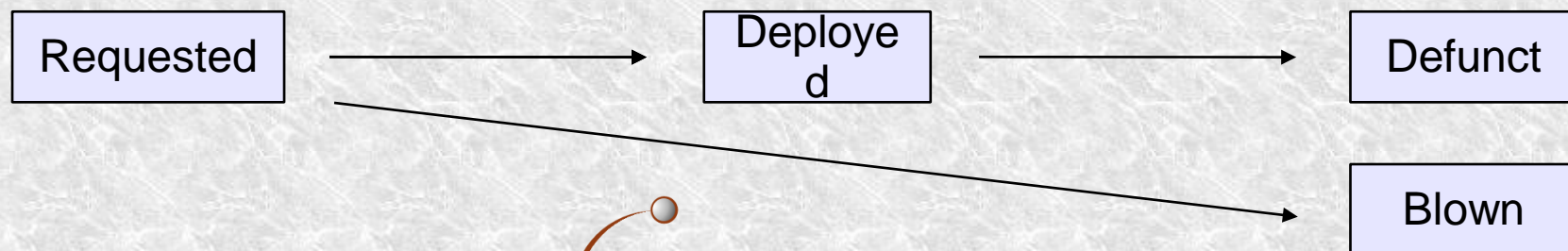
Bundle up

- Put *all* your changes—database or not—into self-contained “bundles.” (zip, war, msi, tar.gz)
- CM the bundles!
- Link these to your tasks. (*Not* CR’s!)
- Bundles have a life cycle:



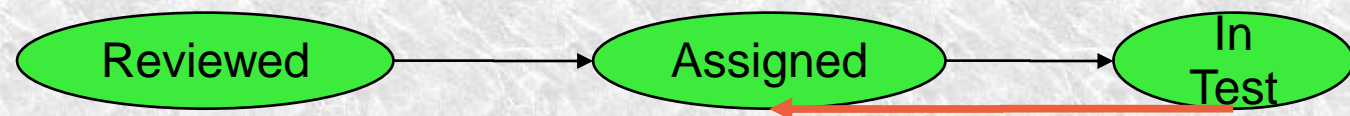
Respect the Deployment!

- Deployment is an activity.
- A deployment *record* is an entity.
- Deployments map bundles onto hardware.
- “Please deploy update 123 to box ‘frodo.’”
- Deployments have a life cycle:



Hardware is not Process

- Too many work flows have server names in them.
 - Would you transition from “IN DEVELOPMENT” to “10.15.131.11”?
 - Nobody outside of development cares that you have a “staging” server.
- You can't equate deployment with process when there's a loop:



LDM Increases Abstraction

- Track “deployment” separately from “status”
 - “on QA server 1 & 2” vs. “In testing”
- More abstraction means a simpler work flow.
 - Use states meaningful to customer, management.
- Simple work flows can be shared.

Enterprise CM

- Different Teams
- Different Environments
- Different Locations
- Different Technologies

Enterprise CM is 'multi-multi' CM

• “The way they were doing it, and the hoops they were jumping through to try and copy this status all over the place, and keep it in sync in the face of multi-everything (multi-site, multi-project, multi-variant, multi-market, multi-release) and having to promote/demote and backup things and backout things, well it was a hairy mess.”

– *Brad Appleton, private correspondence*

LDM + Enterprise CM

- Simpler process
 - + self-contained bundles
 - + abstract “deployment” operation
 - = some multi’s are easy.



Questions

- Thanks for coming.
- Do check the white paper.
- This is what I do;

I will happily talk to you about it
until you are sick of it.