

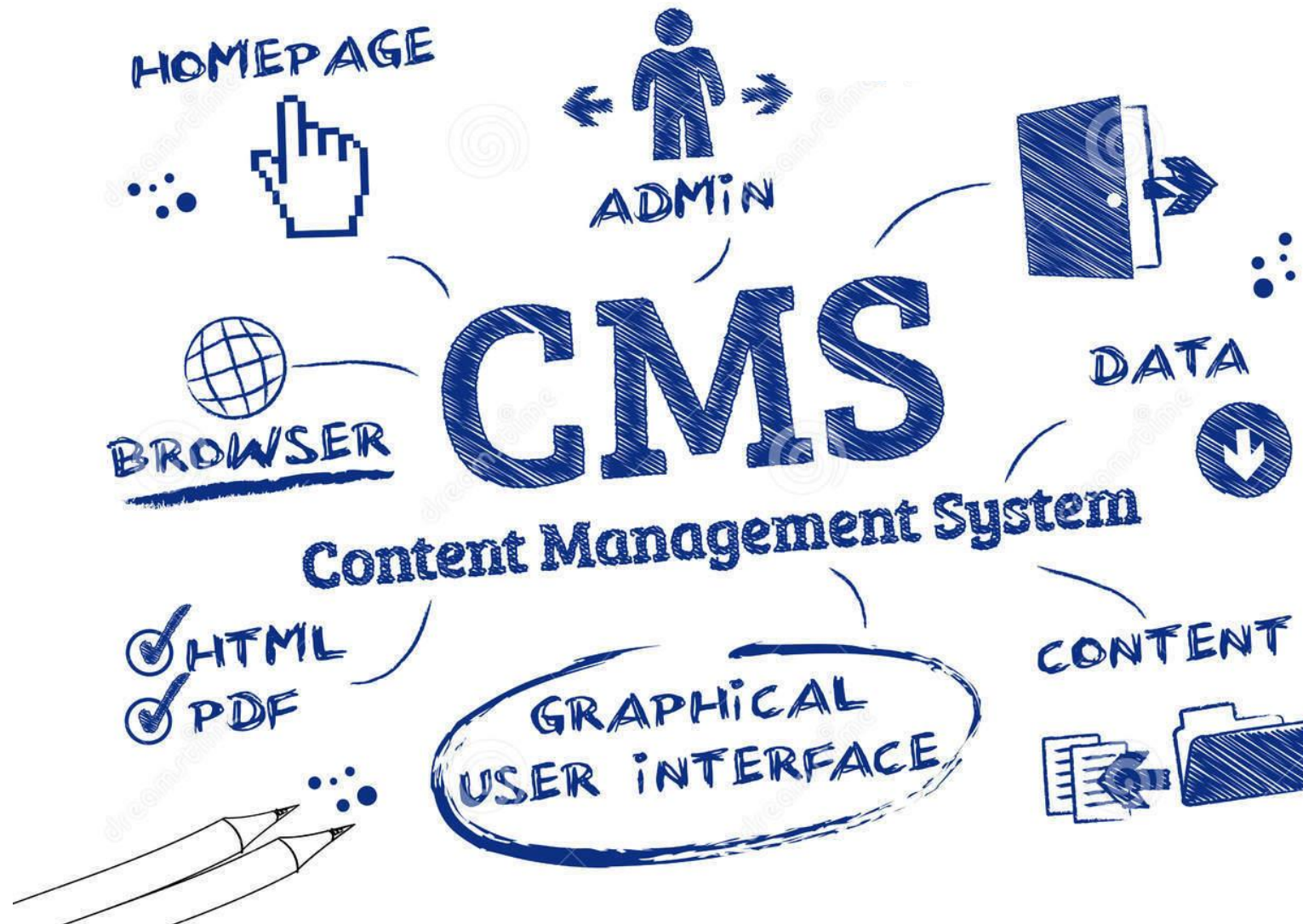
Continuous Delivery and Telerik Sitefinity CMS

Sean Molam
Sitefinity Solutions Consultant
Precedent

Seth Cleaver
Sitefinity Technical Engineer
Telerik



Continuous Delivery and Telerik Sitefinity CMS



Session Agenda

- What is Continuous Delivery (CD)
 - Definition & terms
 - Goals
 - What does a Sitefinity CD environment look like?
 - Potential issues & things to consider
- Hands On
 - Sitefinity & source control
 - Build servers
 - Provisioning
 - Deployment scenarios
 - Content delivery: SiteSync
- Proposed SiteSync Improvements in Sitefinity 8.1

Definitions & Terms

- **Continuous Delivery** is a software development discipline where you build software in such a way that the software can be released to production at any time.
 - Deployable throughout its lifecycle
 - Prioritises keeping software deployable over working on new features
 - Fast, automated feedback on production readiness, any time somebody makes a change to them
 - Push-button deployments: any version, any environment, on demand
- **Continuous Deployment** - every change goes through the pipeline and automatically gets put into production, resulting in many production deployments every day.
- **Continuous Integration** usually refers to integrating, building, and testing code within the development environment. Continuous Delivery builds on this, dealing with the final stages required for production deployment.

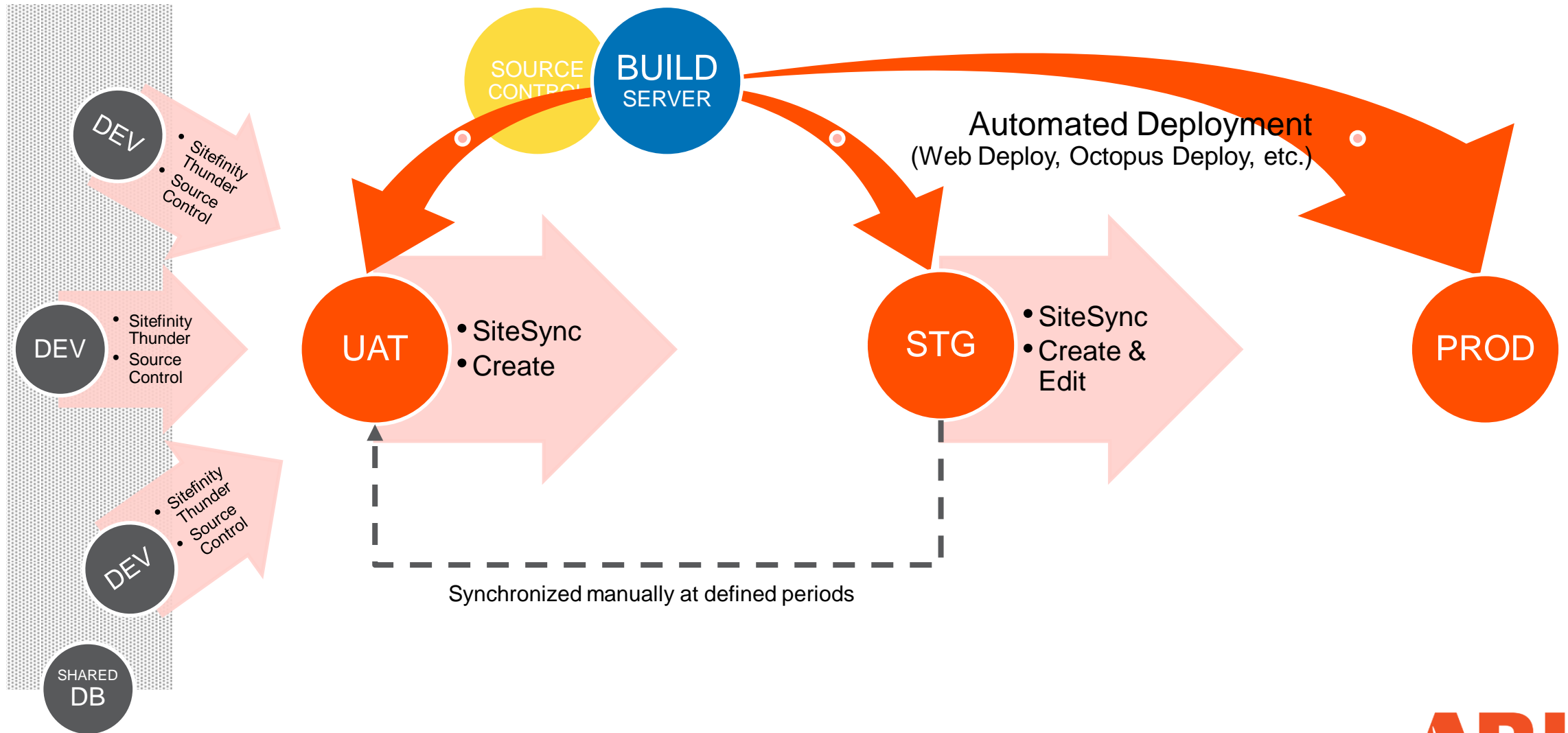
Definitions & Terms

- **Continuous Delivery Pipeline or Deployment Pipeline** is a set of validations through which a piece of software must pass on it's way to release.
- **Continuous Delivery Environment** is a set of processes and tools used to create and test the software product.

Goals

- **Reduced Deployment Risk**
- **Raises the Quality**
- **Perceived Progress**
- **User Feedback**

Continuous Delivery Environment



Potential Issues & Things to Consider

■ Continuous Delivery

- A close, collaborative working relationship between everyone involved in delivery
- Extensive automation of all possible parts of the delivery process

■ Source Control

- Essential part of software development
- Sound source control procedures and policies eg. new branch created for each new feature and ONLY merged with master when ready for release

■ SiteSync

- Establish rules for syncing between UAT > STG as changes on UAT could possible overwrite changes in STG
- Development team needs to be familiar with is and what is not sync'd
- Deploy code changes first, content second
- Scheduled changes are not sync'd, only published ones
- Only one sync job can run at one time

Hands On Demonstration

Sean Molam



Integrating DEC with non-Sitefinity Sites/Applications

Seth Cleaver



Proposed Improvements in Sitefinity 8.1

- Ability to schedule multiple sync operations
- Scheduled publish
- Allow Site Sync to work over https
- Synchronization of unchanged pages
- Implement auto-retry of failed task
- POSSIBLY: Add syncing of configurations
- POSSIBLY: Exclude items from sync
- POSSIBLY: Send email notification when site sync is completed

Q&A Session

Sean Molam & Seth Cleaver



