SharePoint 2010 evolution Conference MMX

Growing SharePoint from small libraries to large scale repositories and massive archives

DD113

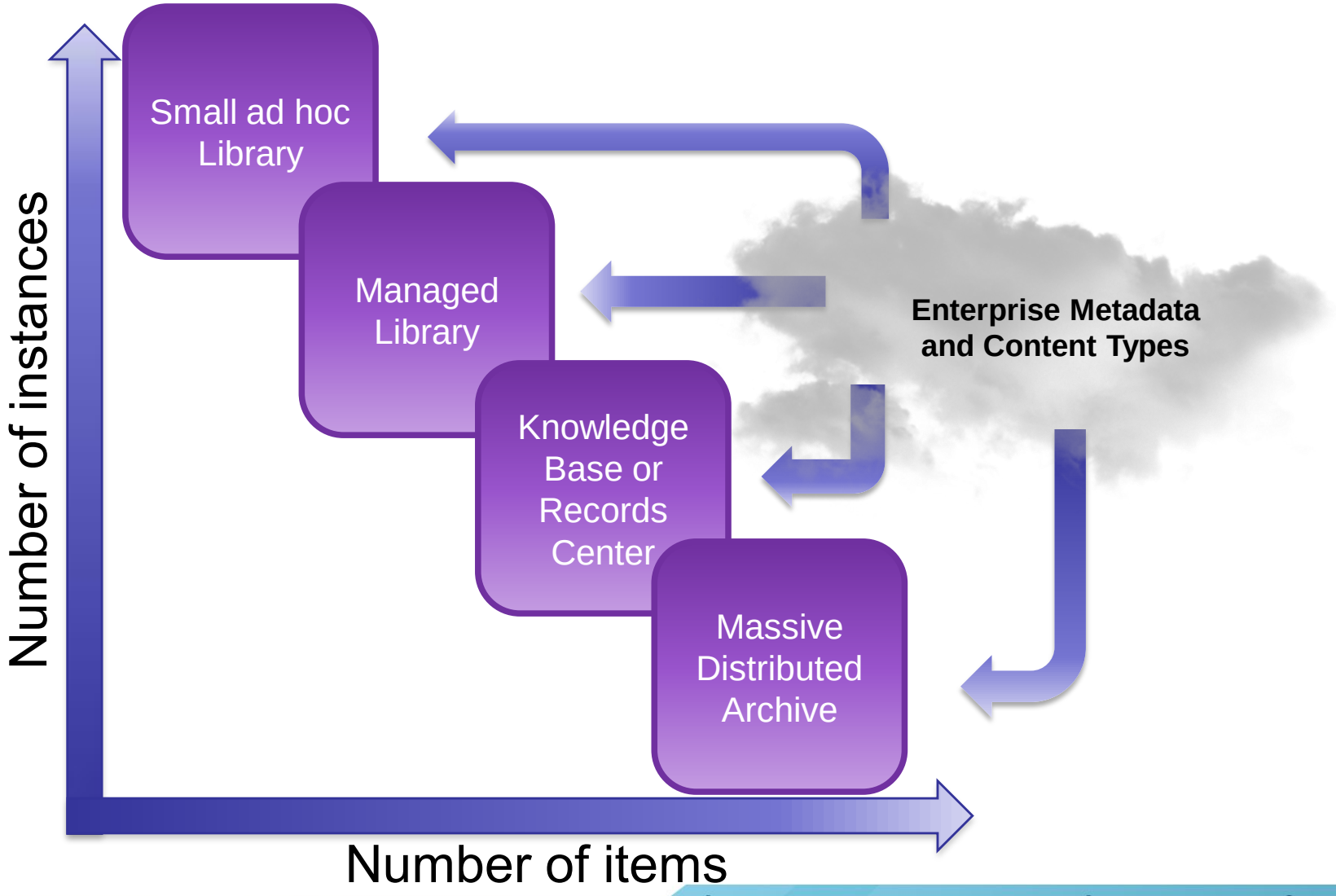
Mirjam van Olst

About Mirjam

- Microsoft Certified Master SharePoint 2007
- MVP SharePoint Server
- SharePoint Architect at Macaw
- Co-organizer DIWUG and SDN

- Blog: http://sharepointchick.com
- Email: mirjam@macaw.nl
- Twitter: @mirjamvanolst

List Scalability / Agenda



Small ad hoc libraries

- Key characteristics
  - Average size: up to 200 docs
  - Used extensively by projects and teams
  - Lots of these created ad hoc throughout the enterprise
  - Easy to create
  - Libraries not managed

Small ad hoc libraries

- Examples
  - Library for storing a management team's work in progress docs
  - A library spun up for collaborating within a particular project
  - A library on your my site for sharing docs with colleagues

Demo

SMALL AD HOC LIBRARIES

Medium sized managed libraries

- Key characteristics
  - Average size: hundreds or thousands of docs
  - Structured/Managed library
  - Managed by one or a few subject owners
  - Possibly one library per department
  - Usually too big to find content without structured/planned navigation
  - Content is important and is reasonably static, so users are willing to file correctly

Medium sized managed libraries

- Examples
  - RFP Response library for a sales force
  - Spec library for an engineering team
  - Brand images repository for marketing
  - Official documents of HR department

Demo

MEDIUM SIZED MANAGED LIBRARIES

Large scale repositories

- Key characteristics
  - Average size: millions to tens of millions of docs
  - Managed by a dedicated team of content stewards
  - Users feel like they are handing off content to the archive
  - Mostly finished work product, meant for broad consumption
  - Virtual folders will help users to find what they are looking for

Large scale repositories

- Examples
  - Corporate records archive
  - Knowledge management repository
  - Centralized best practices repository

List throttling

- Sets limits on how severely users can put the beat down on your servers
- Limits the amount of rows that can be retrieved for a list or library at any one time
- Example:
  - A list with thousands of items
  - A view that would return all items in the list
  - List throttling won’t allow such a request to execute
  - Hit on the server is alleviated
  - User gets a message that his request exceeds the throttle limit for this list

List throttling

- Examples of when list throttling will kick in:
  - Viewing data in a list or library
  - Sorting a large list on a non-indexed column
  - Deleting a web with large flat lists
  - When developing solutions while using for instance SPList.Items (depending on settings)

List throttling - options

- In Web Application general settings:
  - List View Threshold –maximum number of items that can be retrieved in one request. Default: 5,000. Smallest: 2,000.
  - Object Model Override –to enable retrieving items through the object model, up to the List view threshold for auditors and administrators
  - List View Threshold for Auditors and Administrators –only relevant when Object Model Override is yes and only via the object model
  - List View Lookup Threshold – only relevant for lookups
  - Daily Time Window for Large Queries –block of time during the day when list throttling limits are not enforced

List throttling - exceptions

- Scenario 1:
  - Box administrator on the WFE
  - At least Read rights to the list data
- Scenario 2:
  - Using object model SPList class
  - SPList.EnableThrottling = False

Demo

LARGE SCALE REPOSITORIES

Massive distributed archives

- Key characteristics
  - Average size: hundreds of millions of docs
  - Managed by a dedicated team for the archive
  - Content usually gets added by automated processes
  - Lots of process automation to deal with high volume
  - Backend systems, not users, are primary submitter
  - Logical organization and hierarchy is key

Massive distributed archives

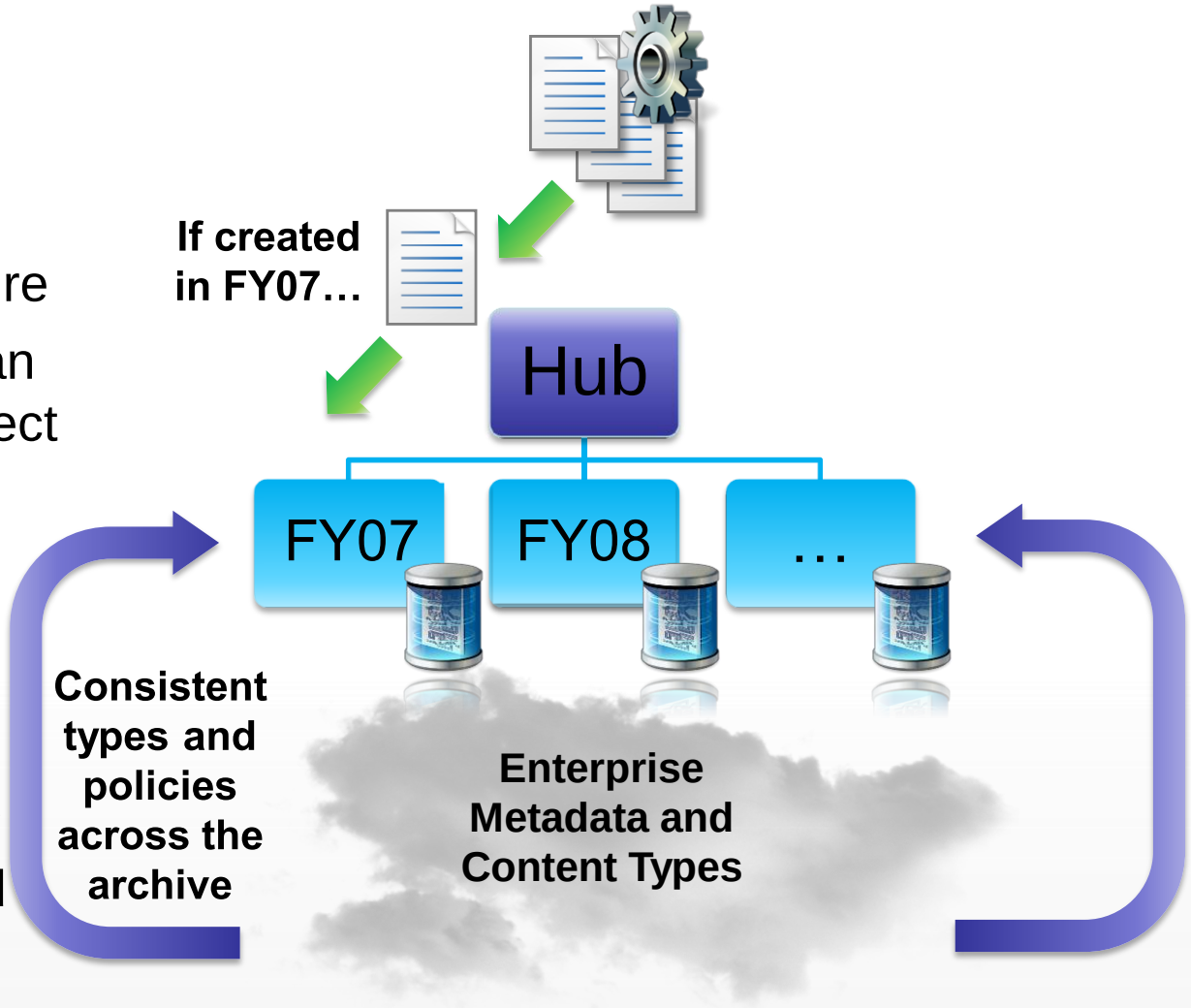
- Essential new features
  - Content Organizer
  - Enterprise Metadata / Taxonomy Term Store
  - Content Type Syndication

Massive distributed archives

- Examples
  - Archive for a large government agency
  - Yearly archive of insurance forms

Massive distributed archives

- Scale is achieved with a distributed architecture
- Content organizer can route content to correct site collection in the archive
- Content type syndication enables central management of distributed archive
- FAST search is used to retrieve content



Content Organizer

- Routing Feature
- Automatically routes documents to different libraries and folders
- Can be used to automatically create new folders after a certain # documents have been added
- No user intervention required
- Does not work for lists, just libraries
- DropOff folder created when “Content Organizer” site feature is enabled

Content Organizer Settings

- Redirect Users to the Drop Off Library
- Sending to another site enables documents to be redirected to other sites, or even site collections
- Folder Partitioning to automatically distribute documents across folders
- Duplicate Submissions to add new versions, or add unique characters to the filename
- Preserving context to keep audit logs and properties

Content Organizer Rules

- Rules primarily based on content types
- Conditions can be added based on document properties
- Target Location can either be a library in the current site, or a different site or site collection
- Target location sites need to:
  - Have the content organizer feature activated
  - Be registered in Central Administration
  - Have the content type available

Content Type Syndication

- Synchronizes content types across site collections
- Select site collection as content type hub in central administration
- Set managed metadata service proxy to consume content types from content type store
- Turn on publishing for content types in hub
- Run timer jobs to make content types available in other sites

Demo

MASSIVE, DISTRIBUTED ARCHIVE

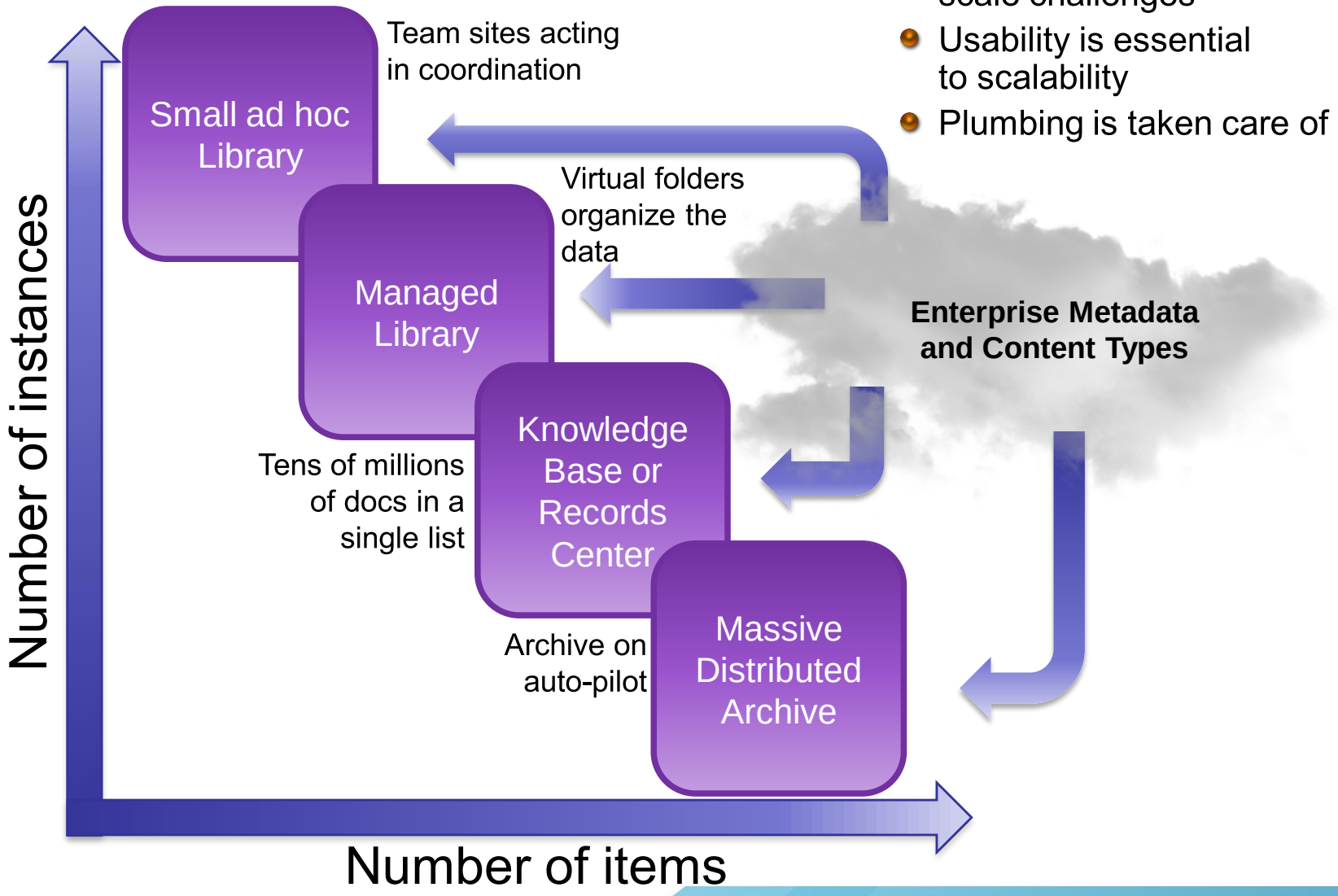
Back End Scale Improvements

- Internal database improvements (e.g. lock ordering, throttling, IOPS efficiency)
- Compound indexing, index management, and content-by-query optimizations
- SQL 2008's Remote Blob Storage (RBS) decreased size of content DB
- Background per-item processing throughput maximization

Wrapping up…

Key Takeaways

- SharePoint 2010 is an **indispensible tool** for data scale challenges
- Usability is essential to scalability
- Plumbing is taken care of



Thank you for attending!