1. Ship Restrictions – BIG DEAL
2. Plan the inventory lookup – JOHN, FARRUKH, JULIA
   1. Monday – service endpoint, strategy to hit it, request, response
   2. Tuesday – what does the front end look like
   3. Wednesday – finish the back end request/response component
   4. Thursday – display the data
   5. Friday – tweak
   6. QUESTION:
      1. What UI are we going to create?
      2. What technologies are we going to use?
      3. What environment is this running in?
      4. What tech are we using?
      5. Where is the inventory service? PO – PM – DEV – TECH LEADS
3. Markdowns – transforming/dropping messages (SNS filters)
   1. Messages to Queues
   2. Marking Item Master
   3. Marking Products in BCC – all good
   4. Removing Tiles
4. Password resets - MONDAY
5. Julia
   1. EU Exports
   2. Documentation?
   3. Inventory Lookup
   4. Support
   5. Feeds for FP – check these today
6. What She's wearing – THURSDAY
7. ~~Comparison thing – JOHN~~
   1. Style number for categories for CN
8. ~~Publishing discussion – spreadsheet to manage case by case basis~~
9. ~~Firewall - change request – EMAILED 3/7~~
   1. Test the firewall change – send all skus to STL DEV
10. CM- china going at 7:15 am
11. UO EU – available product report
12. MoveIt – requested for Farrukh & John
13. UO EU – March 17th
14. TIBCO messages – 7 days message life
15. CA – FTP Creds
16. Employee | Passwords
17. Tile Cleanup

MARKDOWNS CLEAN UP:

1. Query the weblinc data for the freshest markdowns (query at the end of this list)
2. Clear out all the “markdown\_allowed” flags in Item Master
3. Set the markdown flag for every sku in the weblinc data (not short style)
4. Commit changes
5. Run the Process Markdown button in the BCC
   1. Process marks all products as “is markdown” for skus flagged in Item Master
   2. Process reads messages from queue
   3. Process removes tiles from categories
   4. Process marks those tiles as orphaned

TILES CLEAN UP:

1. Query for every tile that is not associated with a category in the BCC
2. Mark the tiles from the query results as “orphaned=true”
3. Commit changes
4. Run the full publish to get the changes out to other systems

MARKDOWN SQL:

select

sss.siteID,

s.shortFulfillmentSystemSku,

p.shortFulfillmentSystemProductID,

sss.retailPrice,

sss.retailSalePrice

from skuSiteSettings sss with (nolock)

join sku s with (nolock) on s.ID = sss.skuID

join product p with (nolock) on p.ID = s.productID

join productSiteSettings pss with (nolock) on pss.productiD = p.ID and pss.siteID = sss.siteID

where sss.siteID <> 1

and pss.doNotDisplay = 0

and sss.applySalePrice = 1

and sss.displayOriginalPrice = 1

and sss.retailSalePrice < sss.retailPrice

AND dbo.skuStatus\_noReserve

(

sss.skuID,

NULL,

sss.dateTimeToStartAllowingPurchase,

sss.dateTimeToStopAllowingPurchase,

sss.alwaysDisplay,

sss.doNotDisplay,

sss.dateTimeToStartDisplaying,

sss.dateTimeToStopDisplaying,

sss.alwaysAllowPurchase,

sss.doNotAllowPurchase,

sss.minimumInventoryToAllowDisplay,

sss.minimumReorderDateProximityToAllowDisplay,

sss.minimumInventoryToAllowPurchase,

sss.minimumReorderDateProximityToAllowPurchase,

sss.availableInventory,

sss.siteID,

sss.reorderDate,

sss.reorderedInventory,

getdate()

) IN ('instock', 'backordered')

order by sss.siteID, p.shortFulfillmentSystemProductID, s.shortFulfillmentSystemSku;