

Class 4: Big Data Application Project Information

New York University
Summer 2017



Project Background Information

- This week you'll create an independent project proposal (IPP)
 - You will use the IPP template provided in NYU Classes to propose a Big Data application
 - Please work on this proposal independently
 - For your IPP, you don't need to write any code, just research data sources and try to formulate a project that interests you
 - Your application should leverage at least two datasets in a way that provides actionable insight and support for actuation decisions (remediations)
- There will be a future opportunity to submit a team proposal

About the Big Data application ...

- Code will be written in Scala and Spark
- If you want to use Python instead of Scala, please let the professor know before you start
- You can incorporate:
 - MLlib - for machine learning
 - GraphX - for graph processing
 - Flume - for data ingest
 - Sqoop - for data input/output to/from a relational database
 - d3 or Tableau - for visualization
 - etc.

Your application will:

- Ingest static and/or changing data (near real time data)
- Process that data to obtain actionable insight
- Suggest an action(s) that should be taken in response to the insight (this is what is meant by an actuation decision)
- In this way, you'll come full circle:

ingestion -> actionable insight -> actuation

This project differs from the Hadoop course project in several ways.

For this project:

1. Use Spark and Spark tools to implement your project
2. Build a complete application - you are encouraged to visualize the results using d3, Tableau, a web UI, a mobile app, etc.
3. Provide one or more crisp actionable insights; the actionable insight(s) you uncover must be crisp insights; it is not sufficient to provide an interface for the user to explore the data and find their own insights
4. This project requires the actuation step

Project Example

Example: Data center energy optimization project with actionable insight and actuation decisions

Temperature sensors were placed densely throughout the aisles of a data center, at 1 ft, 3 ft, and 5 ft elevations. These sensors, and temperature sensors embedded in servers and storage devices, report temperatures into a big data cluster.

- Data was ingested and processed in near real time to monitor the data center and to detect problems
- Prevailing temperatures in all areas of the data center were compared against the normal historic temperature at the same time of day, for a given day
 - This is how we identified and characterized anomalous events, such as a hot spot in a cold aisle
- A hot spot in a cold aisle means that local servers are not being cooled optimally because the cool air intake is receiving warmer air
 - Over time, this will cause the hot spot to grow and more servers to be impacted

Data Center Hot Spot Detection, Diagnosis, and Remediation*

Agents and Emergent Phenomena - Energy Analytics, IBM Research

Joint work with colleagues at IBM T. J. Watson Research Center, Zurich Research Lab, China Development Lab, and Southbury Green Innovation Data Center

* *Not distributable.*

Heatmaps, Hot/Cold Spots for Maximo*

Agents and Emergent Phenomena - Energy Analytics, IBM Research

Joint work with colleagues at IBM T. J. Watson Research Center, Zurich Research Lab, China Development Lab, and Southbury Green Innovation Data Center

* *Not distributable.*

Reducing Energy Consumption in Data Centers

Agents and Emergent Phenomena - Energy Analytics, IBM Research

Joint work with colleagues at IBM T. J. Watson Research Center, Zurich Research Lab, China Development Lab, and Southbury Green Innovation Data Center

- We ran experiments to characterize the effects of various anomalous situations on the prevailing temperature. We found:
 - High CPU utilization may effect temperature
 - Blocking a perforated floor tile prevented cool air from reaching the local servers

- When we compared temperature trends during controlled experiments to historic steady state temperatures, we identified two insights and a remediation for each:
 - **Insight #1:** High CPU load causes a temperature increase of about one degree Celsius in a cold aisle.
 - **Remediation:** Migrate the high CPU job to a cooler area of the data center using virtual machine migration.
 - **Insight #2:** Blocking a perforated tile causes a temperature increase of about 5 degrees Celsius in a cold aisle.
 - **Remediation:** Dispatch a robot equipped with a video camera to the area of the hot spot and capture video of the area which can then be programmatically analyzed for a covered perforated tile.

Autonomously collects data center layout and sensor data

- Obviates laborious manual collection

No prior knowledge of data center needed: just set it down and turn it on

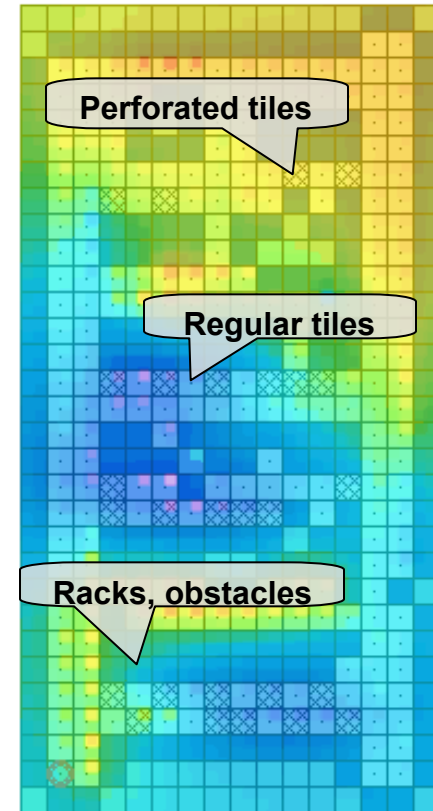
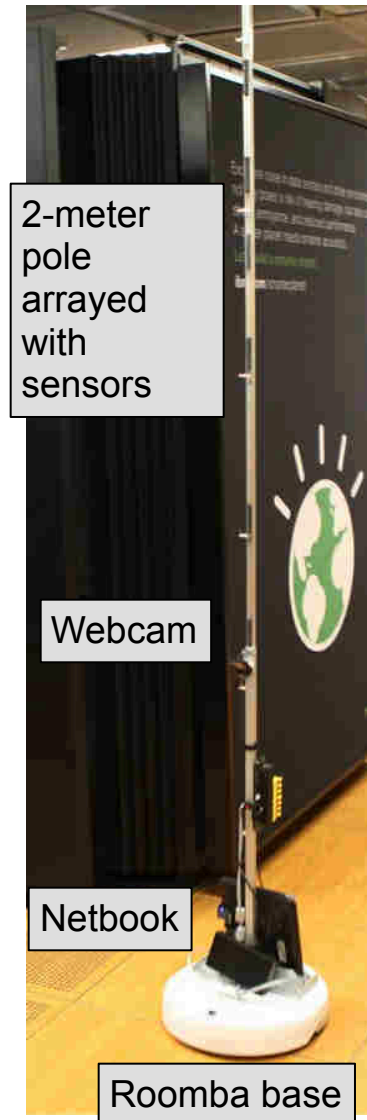
- Uses square tile grid for navigation
- Full data center coverage

Robustly avoids obstacles and copes with collisions

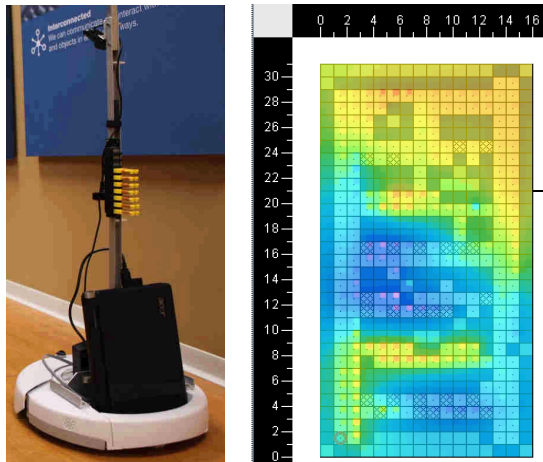
Low construction cost

Provides key support for analytics

- Data are used to generate physics and asset models on which diagnostic and predictive analytics rely
- If left in residence, robot further supports analytics via fine-grained periodic sensing and on-demand remote viewing of trouble spots



Asset layout and temperature map collected by robot at Poughkeepsie Green Data Center



Robotic Temperature and Layout Capture

The robot automates initial data capture required to create physics and asset models on which diagnostic and predictive analytics rely.

Hot Spot Diagnosis

Green Innovation Data Center - High Density Room

The temperature is 4.8 degrees C above normal at location 22.0, 30.0, 3.5

Explore Interactivity with HIC Dashboard...

	Causes	Impact (Degrees C)	Confidence	Remediation Options																								
	Heavy workload <i>*High workload is contributing to the hot spot.</i> <table> <tr> <th>Server</th> <th>CPU Util Curr (%)</th> <th>CPU Util Rise (%)</th> </tr> <tr><td>vmesures01.E3XX</td><td>99</td><td>84</td></tr> <tr><td>vmesures02.E3XX</td><td>100</td><td>91</td></tr> <tr><td>vmesures03.E3XX</td><td>99</td><td>94</td></tr> <tr><td>vmesures04.E3XX</td><td>99</td><td>94</td></tr> <tr><td>vmesures05.E3XX</td><td>96</td><td>84</td></tr> <tr><td>vmesures06.E3XX</td><td>99</td><td>87</td></tr> <tr><td>vmesures07.E3XX</td><td>100</td><td>95</td></tr> </table>	Server	CPU Util Curr (%)	CPU Util Rise (%)	vmesures01.E3XX	99	84	vmesures02.E3XX	100	91	vmesures03.E3XX	99	94	vmesures04.E3XX	99	94	vmesures05.E3XX	96	84	vmesures06.E3XX	99	87	vmesures07.E3XX	100	95	0.4	High	<i>Evacuate workload from these servers...</i>
	Server	CPU Util Curr (%)	CPU Util Rise (%)																									
vmesures01.E3XX	99	84																										
vmesures02.E3XX	100	91																										
vmesures03.E3XX	99	94																										
vmesures04.E3XX	99	94																										
vmesures05.E3XX	96	84																										
vmesures06.E3XX	99	87																										
vmesures07.E3XX	100	95																										
	Perforated tile covered <i>*Perforated tile(s) may be blocked.</i>	1.6	Medium	<i>Send data center robot to check area...</i> 																								


Hotspot Diagnosis and Remediation

- Hot Spot Diagnosis agent
- Merges IT data with facilities data
- Analyzes multiple data sources: ITM, TADDM, Maximo, etc.
- Identifies most likely causes of hot spots; estimates temperature impact and confidence level
- Supports data drill-down
- Remediation technologies such as workload migration and robot dispatch





Hot Spot Diagnosis

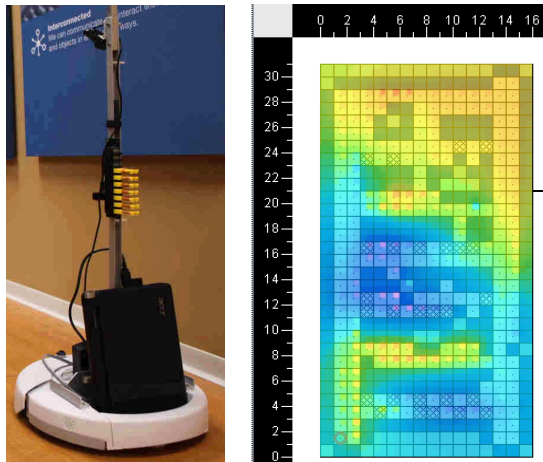
Green Innovation Data Center - High Density Room

The temperature is 4.8 degrees C above normal at location 22.0, 30.0, 3.5



Explore
interactively with
[HC Dashboard...](#)

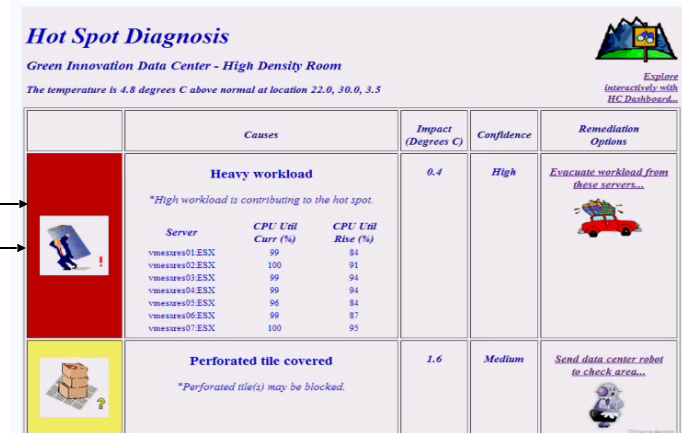
	Causes	Impact (Degrees C)	Confidence	Remediation Options																								
	<p style="text-align: center; color: blue;">Heavy workload</p> <p style="color: blue; font-size: small;">*High workload is contributing to the hot spot.</p> <table border="1" style="width: 100%; border-collapse: collapse; font-size: small;"> <thead> <tr> <th style="text-align: left;">Server</th> <th style="text-align: center;">CPU Util Curr (%)</th> <th style="text-align: center;">CPU Util Rise (%)</th> </tr> </thead> <tbody> <tr><td>vmesxres01.ESX</td><td style="text-align: center;">99</td><td style="text-align: center;">84</td></tr> <tr><td>vmesxres02.ESX</td><td style="text-align: center;">100</td><td style="text-align: center;">91</td></tr> <tr><td>vmesxres03.ESX</td><td style="text-align: center;">99</td><td style="text-align: center;">94</td></tr> <tr><td>vmesxres04.ESX</td><td style="text-align: center;">99</td><td style="text-align: center;">94</td></tr> <tr><td>vmesxres05.ESX</td><td style="text-align: center;">96</td><td style="text-align: center;">84</td></tr> <tr><td>vmesxres06.ESX</td><td style="text-align: center;">99</td><td style="text-align: center;">87</td></tr> <tr><td>vmesxres07.ESX</td><td style="text-align: center;">100</td><td style="text-align: center;">95</td></tr> </tbody> </table>	Server	CPU Util Curr (%)	CPU Util Rise (%)	vmesxres01.ESX	99	84	vmesxres02.ESX	100	91	vmesxres03.ESX	99	94	vmesxres04.ESX	99	94	vmesxres05.ESX	96	84	vmesxres06.ESX	99	87	vmesxres07.ESX	100	95	0.4	High	<p style="color: blue; font-size: small;"><u>Evacuate workload from these servers...</u></p> <div style="text-align: center;">  </div>
Server	CPU Util Curr (%)	CPU Util Rise (%)																										
vmesxres01.ESX	99	84																										
vmesxres02.ESX	100	91																										
vmesxres03.ESX	99	94																										
vmesxres04.ESX	99	94																										
vmesxres05.ESX	96	84																										
vmesxres06.ESX	99	87																										
vmesxres07.ESX	100	95																										
	<p style="text-align: center; color: blue;">Perforated tile covered</p> <p style="color: blue; font-size: small;">*Perforated tile(s) may be blocked.</p>	1.6	Medium	<p style="color: blue; font-size: small;"><u>Send data center robot to check area...</u></p> <div style="text-align: center;">  </div>																								



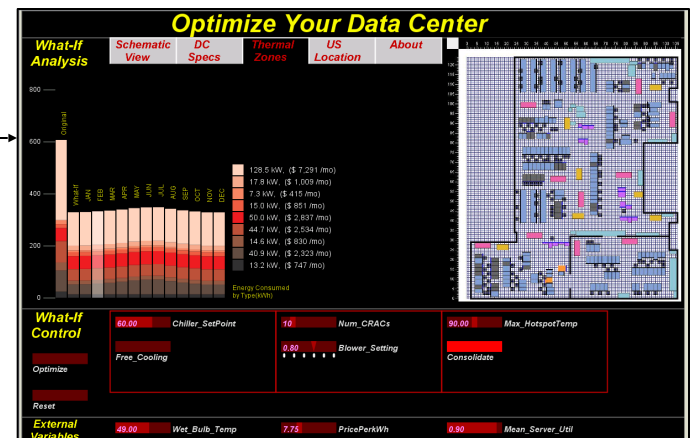
Robotic Temperature and Layout Capture

The robot automates initial data capture required to create physics and asset models on which diagnostic and predictive analytics rely.

If left in residence, the robot can further support diagnostics and predictive analytics with periodic or on-demand monitoring.



Predictive energy analytics assists diagnosis and can support preview of remedial actions.

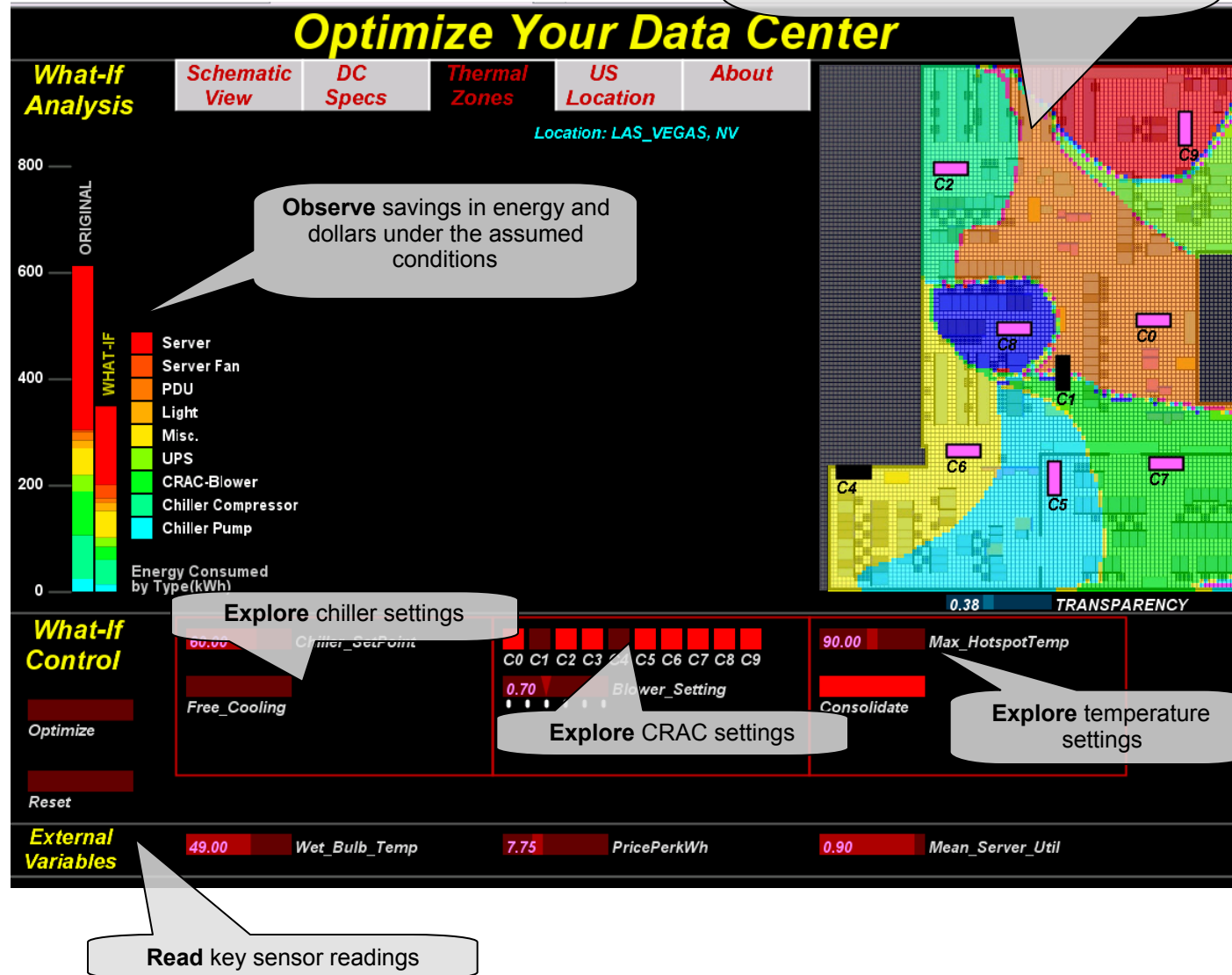


Predictive Energy Analytics

Big Data Application Development

Predictive Energy Analytics

- Performs what-if analysis on annual or instantaneous basis
- Estimates annual energy savings from about two dozen easily-gathered inputs
- Takes advantage of asset layout details to generate more refined estimates and provide dynamic operational guidance



Steps for Developing your Application

Steps for Developing your Application

Identify a problem that your application will solve

- Formulate the problem to be solved and then try to find supporting data
- ***Alternately***, you can first assess the data that is available to you and then identify a problem that can be solved using the available data

Steps for Developing your Application

Expertise?

- Whatever problem you choose to solve, be sure that your team has knowledge of the problem space, is willing to learn about it, and/or will confer with one or more experts

For example: You want to develop an application that predicts when a drought will occur and also provides remediation.

- Questions:
 - Do you know anything about *drought prediction*?
 - Does anyone on your team have this background?
 - Do you know how to avoid drought?
 - Do you know what actions to take in case of a drought?
 - Do you know where to find needed information?

Steps for Developing your Application (continued)

Questions to Answer

1. Who benefits from your application?
2. Who is the user of your application?
3. How can you be certain of the **goodness** (correctness) of your application?
 - By comparing to known outcomes?
 - Using historic data?
 - Finding published material that matches your outcome?
 - Finding published material that doesn't matches your outcome, and explaining why?

Steps for Developing your Application (continued)

Obtain data

- What types of data do you need for your application?
- How hard or easy will it be to obtain the data that you need?
- Is there open data available on the internet or by request?
- Is the data that you need privately held? Will you be permitted to use it?
- If you need to request access, it is best to send an email as soon as possible. It is better to have permission to use data even if you wind up not needing it.
- Is the data very sensitive, such as patient medical data? If so, it may be possible to obtain the data if the data owner is willing to pre-process it to anonymize it. This type of data is often very hard to obtain.
- Can you identify alternative data sources and approaches in case the data you need cannot be obtained?
- How much cleaning and reformatting of the data is required in order to make it useful for your application?

Steps for Developing your Application (continued)

Where to find data?

- You can find data (and ideas) by exploring open data sources provided at federal, state, and local government levels (e.g. NYC 311 data, police data, etc.)
- Please go to <http://datausa.io/>
 - This is a website that visualizes several open datasets
 - Click on About and read about Data USA (link: <http://datausa.io/about/>)
 - Click on Data and explore the data sources
 - Read: <https://github.com/DataUSA/datausa-api/wiki>
- Also explore the datasets you find here: <https://www.data.gov/>
- Some questions to think about:
 - How do you like the visualizations? What would you change? Do the visualizations combine datasets, or do they just focus on one dataset?
 - How can the data be accessed - via download? API?
 - How much data is there in these datasets?
 - How often is the data updated? Does past data get modified, or is new data simply appended?

Steps for Developing your Application *(continued)*

Consider Feasibility

- Can your application be developed by the end of the semester?
- Do you have enough team members to support the application you want to build?
- Are the collective skills of team members sufficient to support the application you want to build?
- Can you obtain the required data in time to use it in your application?

Steps for Developing your Application *(continued)*

Select programming technologies

- Which Spark technologies will you use in your application?
- Do you need to install these technologies or are they already available in your Hadoop environment?
- For technologies not covered in class, does your team have enough time and sufficient background to learn those technologies?

Steps for Developing your Application (continued)

Pre-process the data (Know Your Data)

- Write code to profile the data
 - What is the total number of records in each data source?
 - What are the maximum and minimum values for each column?
 - If the column contains free-format text, what is the longest line? What is the shortest line? If the same text repeats, can you transform it by assigning a code instead of repeatedly storing the text?
 - What is the range of values for each column?
 - How many times does a given value appear in the column?
 - What is the average value for each column (if applicable)?
- Write this code using Spark - ***do not use code that doesn't scale!***
 - ***For example, do not use straight Scala, Java, Python, C++, C, Ruby, ...***

Steps for Developing your Application *(continued)*

Get early results

- Try to find out **early** if your thesis will hold by analyzing a subset of the data
- Early testing will help you identify **and address** problems with the data or the thesis
- Iterate on your approach until your goal is achieved
 - Be flexible, you may need to adjust your approach several times before you arrive at a solution that yields good results

Steps for Developing your Application *(continued)*

Iterating over your application ...

- A failed thesis is the first step in the evolutionary process, don't worry, a better approach evolves
 - It may require re-design
 - It may require addition of one or more data sources (common)

Steps for Developing your Application (continued)

Analyze the results (Assess the goodness of your application)

- *Look critically at your results – your thesis may or may not hold.*
- If your thesis *did not* hold, identify the root cause –
 - Which assumptions were proven wrong? Why?
 - You are the expert on this application - what would be your next step(s) if you had time to continue working on the application?
 - Would the addition of one or more data sources be helpful?
- If your thesis *did* hold, *assess the goodness of your application* -
 - Compare your results to known results
 - For example, if your application predicts stock prices, run your application on historical data and compare your predicted price to the actual, known price.

Join a team!

- Teams have up to three members (members must be enrolled in the course, cannot be auditors)
- Each team member will select one data source that they will learn about, ingest into HDFS, clean, format (Extract-Transform-Load - ETL), process
- The data sources must be combined to produce an application
- It is recommended that teams having just one member use at least two data sources in order to produce an application that has depth and is interesting (but one data source is acceptable for a one-person team)
- To manage the work, split it up among team members
 - An 'agile' approach is a good way to go – following this approach to software development means that you are always striving to have something demo-able at the end of each sprint

Summary - Steps for Developing your Application

1. Formulate a general idea of the type of application to be developed, what are you trying to solve?
2. Learn about the data that's available to you.
3. Request access to the data – do this early, request access to multiple data sources, even if you wind up not using all of them.
4. Identify an application that can provide one or more crisp actionable insights and clear remediation steps.
5. Clean out the data to remove irregularities (sometimes there are hiccups in the data).
6. Identify the right Spark technologies to use.
7. Develop the first version of the application.
8. *Look critically at your results.*
9. *Iterate* on your approach until goal achieved.
10. Assess the goodness of your application - do you trust that the insights obtained are correct, and are the remediation steps reasonable.

Take a Risk!

- This is key!
- Follow the process, start getting data sources ***early***
- Aim high
- The penalty for not taking a risk is higher than for trying and falling a little short
- With time and effort, issues can be overcome

What kinds of applications can we consider??

Computing

Internet of Things (IoT) <http://asmarterplanet.com/blog/2012/12/22400.html>

Robotics

Linked Data

Community Service

Green Energy Initiatives

Meals on Wheels

World Health Organization (United Nations WHO)

Personal Safety

Jobs

Sustainable Energy

Recycling

Renewables

***NYU Green Grants**

Solar Energy

Green Energy Initiatives

Potable Drinking Water

Smarter Solutions

Smarter Buildings

Smarter Infrastructure

Smarter Transportation

Smarter Security

Smarter Datacenter

Politics

Election

Hot topics

Peace

***NYU Green Grants info - <http://www.nyu.edu/sustainability/campus.projects/greengrants/index.php>**