Exhibit No. ___(JMK-2)

BEFORE THE WASHINGTON UTILITIES AND TRANSPORTATION COMMISSION

DOCKET NO. UE-15________

DOCKET NO. UG-15________

EXHIBIT NO._____(JMK-2)

JAMES M. KENSOK

REPRESENTING AVISTA CORPORATION

Exhibit No.__(JMK-2)

Revised Timeline and Budget Forecast

Avista’s Project Compass

Avista Utilities

June 2014

Exhibit No.__(JMK-2)

Avista’s Project Compass

Revised Project Timeline and Budget Forecast

Q. Why is the Company revising its initial project plan?

A. Avista is in the latter stages of implementing its new Customer Service and Work and Asset Management software systems, named “Project Compass” (or “Project” or “System”). The Company is installing Oracle’s Customer Care & Billing system (or “CC&B”), and IBM’s Maximo Work and Asset Management system (or “Maximo”). The initial Project plan was completed in 2012 and envisioned a launch of the new System, known as the “Go Live,” in Q3 2014. Through the course of implementation, the Project team has developed much-more complete information about the full detail of the System work requirements and its ultimate cost. This information, which is described below in this report, provides the basis for the current revision of the initial plan. The overarching consideration for revising the schedule is ensuring the new computer applications undergo thorough testing to validate they will perform at a level, when launched, to execute critical business functions properly and minimize the potential for disruptions to our customers and the Company. The Compass management team determined a Q3 Go Live would not provide sufficient time for the robust testing needed to ensure the readiness of the new applications. Accordingly, the Company’s officers recently agreed to extend the Go Live time frame to include Q1 2015.

Q. Did the Company’s plan and schedule, as initially developed, provide adequate time for testing the System?

A. Yes. The initial work plan generally provided ample time for comprehensive application testing. But, because there were longer than estimated delivery times required by several implementation activities, the new System was not ready to commence testing on the schedule originally envisioned.

Q. Specifically, what work processes took longer to complete?

A. The key activities that required additional time were the development of code for “Extensions” to the CC&B application, and the currently-ongoing process of “Defect Management” associated with application testing. Secondary activities that required additional time, included “System Configuration,” writing “Test Cases” to support the testing protocol, the processes of “Data Conversion” for both CC&B and Maximo, and the development of “Integration Code” for the new replacement System and interconnected applications and systems.

Q. Please briefly describe each of the work processes mentioned above?

A. System Configuration – “Configuring” an application is the process of setting parameters in a vendor’s computer software that enables its built-in logic to perform the functions required by the Company’s various work processes. The process involves selecting among options, embedding algorithms, entering data, and creating specialized instructions. Configuration is performed through a series of input tables that organize the process of setting parameters. Each input table, which could represent one particular type of customer service agreement, for example, may have up to 100 individual, flexible, and configurable fields. Configuring each field requires entering from one to several individual values, instructions, or algorithms to establish the new base System. Each field in each table is often cross-linked with content in dependent fields in complementary tables, creating a complex of dependencies between many multiples of tables and fields. This initial work requires the person entering the configuration settings on a particular table to work iteratively and sequentially in configuring the dependent fields in the other tables as one integrated work flow. As one example of the work involved, it required one technician working full time over six months to configure Avista’s existing rate tariffs into CC&B (142 different service agreements across our three jurisdictions). Considering that CC&B has 1,686 configuration tables, containing 12,158 configurable fields, the magnitude and complexity of this task is quickly evident.

Extension Code – There is considerable flexibility to accommodate a range of business processes within the application's off-the-shelf Configuration settings. But, many business steps are complex enough that they require programming of specialized software code that is outside the application itself. The capability enabled by this specialized code is referred to as an application "Extension." The process of developing this code, which is complex and labor intensive, begins with a description of the work process steps that a particular extension will perform (its technical requirements). Each set of requirements is then translated into a technical specification that guides development of the actual programming code. Once the technical staff has written the code, it is subjected to several iterations of "Unit Testing." Unit Testing validates that the unit of code, in isolation from the System, properly performs the steps identified in the technical specification.

Integration Code – "Integrations" refer to the connections between separate computer applications that allow them to work in concert to perform allied functions. An integration may involve exchanges of data, transmission of instructions or changes in state, performance of computations and other algorithms, and myriad other shared functions. Like Extensions, Integrations require the development of specialized programming code that connects the CC&B application with the Maximo application, and that connects them both with the approximately 100 other applications and systems required to support the Company's customer service and business operations. Some of these systems include the Avista customer website, the Company's various internal systems (such as financial applications, varied databases, supply chain, crew dispatch, outage management reporting), systems of outside financial institutions used by the Company and our customers, and the many vendors who support our delivery of natural gas and electric service, such as bill printing and presentment. In addition to Integration connections between applications, this work also encompasses the development of Avista's "enterprise service bus." The latter is essentially an Integration network that is shared by the integrated applications. The process of developing and Unit Testing the Integration code mirrors that of the code for Extensions, described above.

Code Defect Management – The work of Configuration and coding Extensions and Integrations is very complex and highly interrelated. As a consequence, it is inherent that each unit of the completed work will require several iterations of testing and modification before it will properly execute its part of a business process. Portions of the configuration settings and the specialized code, which initially do not perform properly, are known in the industry as "Defects." Defects are identified during testing when the configured application and specialized code are run through a simulated business process referred to as a "Test Case." During the test, the program simulation runs to the point where a Defect is encountered and the simulation is halted. In the work process known as "Defect Management," that Defect is located and analyzed, and is returned to the Configuration or coding team for correction. The revised code is then run through the very same test-case simulation until the next-limiting defect is encountered. This process is iteratively repeated until all of the defects in that unit of code or Configuration, for that one unique Test Case, have been located and repaired. Then, the testing process is repeated for the next individual Test Case. Over a cycle of testing, it is typical for the rate of defects to be relatively low, initially, and then to increase to a peak before tapering back down to a low and predictable rate. This pattern is important because during the initial testing it is impossible to predict the ultimate number or complexity of Defects in a unit of code. Only at the point where the number of Defects peaks and begins to decline in a predictable way can the remaining Defect-Management effort be reliably forecast.

Application Testing – Three major areas of testing play a critical role in the successful implementation of the new applications. Each type of testing is associated with its own unique process of code Defect Management. “**System Testing”** commences when the work of Configuration and the coding of Extensions is complete. Its purpose is to ensure the new applications perform properly as they have been Configured and coded to support Avista’s business processes. “**Systems Integration Testing”** occurs next in the sequence and focuses on testing the specialized Integration code to ensure the new applications perform properly with all of the other integrated applications and systems. This is followed by “**User Acceptance Testing,**” which is performed by Avista employees who will be using the new System to serve our customers. It has the twin objectives of scrubbing the System to further identify and repair any critical Configuration, Extension or Integration Defects, and to identify and implement changes to the System that will make it more user friendly and function more smoothly and efficiently for customers and employees.

Simulation Test Cases – Test-Case scenarios are written to evaluate virtually every step of every business process that is enabled by the new System. Each Test Case is unique from all other Test Cases and is written to evaluate a very specific portion of the configured application or specialized code. The complexity of the applications requires a significant number of unique Test Cases to fully validate the integrity of the new System. The number of Test Cases written for each phase of testing of the Company’s new applications, is presented below.

Application Testing	Number of Test Cases
Avista Utilities’ Customer Web Portal	1,283
CC&B Credit and Collections System	667
CC&B Credit and Collections System Integration	407
CC&B System Test	1,472
CC&B System Integration Test	2,471
Maximo System Test	210
Maximo System Integration Test	454
Interactive Telephone System Test	351

Total	**7,315**

Data Conversion – All of the Company's existing data, whether customer account information, energy-use history, electric and natural gas facilities data of all types, mapping system information, and regulatory and compliance information, etc., must be transferred from existing computer hardware and data bases, such as the Company's current mainframe platform, to new data formats, databases, and computer platforms connected to the new applications. To accomplish the conversion, data in the existing databases is mapped according to where it will eventually reside in the new databases. The data are then extracted from the old databases, are transformed as necessary, and are loaded into the new databases. The integrity of the loaded data is then validated for accuracy. Defects in data conversion are identified in the process, Defects are repaired, and the data load/validation exercise is repeated.

Q. Why are these work processes taking longer to complete than was initially planned?"

A. The longer implementation times are primarily the result of the high degree of complexity of the integrated systems being installed by the Company.

Q. What do you mean by "complexity of the integrated systems?"

A. While it's common for a business to install one major system at a time, such as a customer service, financial management, supply chain or asset management system, the Company is installing two major systems simultaneously (CC&B and Maximo Asset Management). Avista is required to implement both new applications because our legacy System contains a customer service module and work and asset management module that are highly integrated, mainframe based, and both in need of replacement. As described above, this effort requires not only that these two systems be custom integrated, but that together, they be integrated with the approximately 100 other applications and systems required to perform the Company's integrated business operations.

In addition to the number of other applications and systems, Avista has several complex applications that many utilities do not possess. Some of these include our Avista Facilities Mapping system ("AFM"), which geographically displays every element of our electric and natural gas facilities in a Geographic Information System (GIS) map format; our Outage Management System, which integrates outage management computer logic with the AFM system to provide accurate outage information for customers and diagnostic tools that reduce outage restoration time and costs; and our Central Dispatch System, which integrates AFM, the Outage Management System, and our Mobile Workforce Management application, to optimize the dispatch and management of restoration crews in real time across our entire electric and natural gas system.

The degree of complexity of the new System is also impacted by the diversity of service provided by the utility. Because Avista provides both natural gas and electric service, the complexity is substantially greater than that of a utility providing either one or the other. Further, the Company provides service in three regulated jurisdictions, each of which has separate and unique operating tariffs and rules that must be coded into the new applications. For portions of our new System, Avista's application configuration and specialized coding will be roughly five times greater than that of a single-fuel utility operating in one state.

Q. Did Avista take steps to understand the source of and to mitigate the impact caused by the longer code development?

A. Yes it did. In December 2013, the Project Compass team assessed the relationship between the complexity of Avista's code requirements, the project schedule, and the level of staffing applied to the work. The end result was that Avista's integration contractor retained additional resources to bolster its overseas code-development team. Progress on the other activities that were taking additional time (application configuration, data conversion, integration code, and writing the test cases) was managed to ensure that applicable portions were ready for System Testing once the CC&B Extension code was available. Through this analysis and actions taken, the Company believed it could better manage the overall time required for coding extensions.

Q. Why didn't the Company change its forecast of the Go Live date earlier in 2014?

A. The Project Compass team concluded that even with an expected addition of time for code completion, that it might be able to make up the time and maintain a Q3 Go Live. The team specifically investigated the structure and schedule allotted for testing the new System, as the primary tool for managing the overall Go Live schedule. The Company wanted to test these ideas before making any formal decision to revise the schedule.

Q. How did the team propose to change its testing protocol in an effort to maintain its initial Go Live schedule?

A. As described above, the System Testing, System Integration Testing, and the User Acceptance Testing, are typically performed in sequence. Each phase of testing, including the process of Defect Management, is relatively complete before the next phase is initiated. The Project Compass team revised this testing protocol to partially overlap the phases of testing to be conducted. In this approach, completed "portions" of an application are subjected to limited System Testing and then to limited System Integration Testing with similarly-completed portions of the other application, including the required Integrations. The net effect of this testing protocol, if successful, would be a reduction in the overall calendar time allotted to application testing.

Q. What did the Project Compass Team learn from the overlapping testing approach?

A. The Company implemented and evaluated this approach for System Testing and concluded that it did reduce the time required for this test phase. But, because of the emerging complexity and additional time required for code Defect Management, the overlapping testing was not able to sufficiently reduce the time required for a successful Go Live. Because overlapping testing adds complexity, and because code Defect Management was becoming the more critical scheduling constraint, the team has made limited use of the overlapping testing protocol for the System Integration and User Acceptance Testing.

Q. What impact is Defect Management having on the overall Project schedule?

A. Avista has experienced greater complexity with the Project Compass Defects than had been anticipated. The result is that even though some time was saved by overlapping portions of the System Test, it has been offset by additional time being spent on Defect Management. The result is the present revision of the overall Project timeline to include Q1 2015.

Q. What steps has Avista taken to reduce the time being spent on code Defect Management?

A. Avista has implemented actions in the areas of process cycle time and testing protocol to improve the rate, or velocity, of Defect repair.

<u>Process Cycle time</u> – Avista worked with its system-integration contractors to reduce the time required for defects in the code to be repaired by the development team and returned to Avista for the next round of testing. Actions have included changing communication protocols, assigning key development staff of the contractors to work from Avista's offices, and modifying schedules of the overseas development teams.

<u>Testing Protocol</u> – In a conventional testing protocol, as described above, the Test Case scenario will be run until a limiting Defect is encountered. The testing is then stopped,

Exhibit No.__(JMK-2)

the Defect is located and analyzed, and it's returned to the development team for repair. The Company is piloting a revised protocol where an identified Defect is patched with a temporary workaround, and the Test Case is continued until the next-limiting Defect is encountered. When possible, the second Defect is likewise patched, and testing is continued until the point where a limiting Defect blocks any workaround and further testing. Then, these accumulated Defects are analyzed and sent to the development team for repair. The intent is that by aggregating several Defects at a time it will improve the overall velocity of code Defect Management.

Q. What additional steps has the Company taken to help control the overall Go Live schedule?

A. The company has implemented changes to the Data Conversion process for CC&B and Maximo. These have helped accelerate Data Conversion and have improved the efficiency of the data validation process. Additional project resources have been added to various workstreams such as the Customer Web Integration effort. System-integration contractors have arranged for their lead staff to spend additional direct time with Avista's team in Spokane, and Avista employs a fifty-hour work week, as needed, to meet peak Project demands. The Project team has also increased the capability of the computer systems supporting the application testing processes. This allows the iterative Test Cases to be run more quickly, further accelerating the Defect Management process. In addition, the Test Cases are being re-prioritized to help ensure the most important business processes are tested and repaired first. The team has also launched the first wave of training for its customer service employees who will be using the new CC&B application. Finally, the Project managing directors are working to ensure morale of employees and contractors remains at a high level for the intensive duration of the Project.

Q. Has the revised implementation plan impacted the Project budget?

A. Yes. The longer time frame required to complete the work processes described above are in large part responsible for the addition of approximately $18 million to the estimated Project budget. This additional capital budget amount, forecast by cost category, is presented in the table below.

Compass Major Costs	**$(1000's)**
System Integrators	**$3,163**
Avista Labor / Loadings	**$4,661**
Technology Contractors	**$3,201**
AFUDC	**$3,609**
Software Licenses	**$480**
Common (PMO)	**$654**
Hardware/Hosting	**$10**
Oracle DB License	**-**
Contingency	**$2,150**
Total	**$17,927**

The revised capital budget authorization for Project Compass is $100 million, which was approved by the Company's officers and Board of Directors on May 8, 2014.

Q. When you say "in part" do you mean there are other factors driving an increase in the project budget beyond a later implementation?

A. Yes. There have been a number of additions to the Project that have contributed to its overall cost, and that were not known at the time the Project plan and budget were assembled in 2012. These changes to the implementation of the applications have been tracked through a formalized process known as a "Project Change Request." The sum of these changes represents a total cost addition of $9.128 million.

Q. Can you provide some examples of the activities and costs that comprise these Project Change Requests?

A. Yes. One of the larger cost items (approximately $1.8 million) is associated with the Company's AFM system. During implementation, the Compass team learned that a GIS software update would provide for a more efficient transfer of data between the AFM system and the new Maximo and CC&B applications. Another addition to the Project was the development of a more-comprehensive customer communication plan (approximately $1 million) to precede the Go Live of the new System. The plan includes ad placement and a direct mailing that identifies subtle changes and improvements in service, as well as the potentially-longer service times (such as call hold time and average time per call) that are expected to temporarily coincide with the Go Live of the new System. Another substantial addition to the capital cost of Project Compass was the inclusion of software maintenance fees to cover the second year of implementation (approximately $998,000). Most of the Project Change Requests have addressed the need for additional technical resources to accomplish specific tasks during implementation of the new systems. For a brief description of each of these Project Change Requests please see Attachment A to this report.

Q. Didn't the Company have a "contingency" in its initial budget to accommodate such changes?

A. Yes. The $80 million initial capital authorization included a contingency amount of $7.176 million. This contingency has offset the majority of the costs added through Project Change Requests.

Q. Has the Company established a definitive date for the Go Live?

A. Not at this point. While the Project Compass team believes that a Go Live window that includes Q1 2015 will provide sufficient time for an effective implementation of the Project, it must complete the bulk of the testing and Defect Management processes before it has confidence in setting a definitive date. When the Go Live date has been selected it will be shared with customers through the communication plan.

Q. Does the Company believe the Project Compass Costs, including the budget additions, are reasonable and prudent?

A. Yes. The original timeline and budget were important project management tools that, while much more refined than the earliest estimates, were still associated with some degree of uncertainty. As described above, when the initial estimates of time and resources required for coding the extensions were developed, the team had no way of knowing the precise degrees of complexity of the coding, the resources required to meet a specified timeline, or the degree of complexity of the defect management process. If the Project team had that precise foreknowledge, it may have added resources and budget to the Project to achieve the initial Go Live date, or it may have added budget to the initially-planned resources to achieve a later date. Because the Project is costing more to implement than was initially estimated, doesn't mean it is no longer the least-cost solution for our customers. Avista believes its revised implementation plan and budget simply reflects a more accurate assessment of the true cost of implementing the Project.

Q. How does the Company believe the implementation of large IT projects should be evaluated?

A. First, Avista is not aware of any large enterprise application system that has been installed by a peer utility that explicitly achieved its initial estimates of timeline and budget. That said, there are distinguishing factors in every project that are useful in helping to assess the reasonableness of its costs. In extreme cases, some companies have abandoned the applications during the course of implementation; the new systems are never placed in service. These failures are often followed by an entirely new selection and implementation effort. In less dire cases, the company may learn during the course of implementation that it selected a less than optimum solution set, which requires a significant and expensive workaround to successfully install. In some cases, the scope of functionality has been set either too broad or too restrictive. In either case, the costs and the time delay associated with mitigating those initial choices can be very substantial. In other cases, companies have made implementation errors such as overlooking basic required functionality, resulting in additional time and budget to include while the majority of the project is awaiting the Go Live. In the best cases, companies have simply underestimated, to varying degrees, the true cost of implementing the selected applications. In other words, these companies have completed a comprehensive needs assessment, prepared a balanced project scope, conducted a robust selection process, selected the proper solutions, hired capable implementation contractors, adequately prepared their organizations for the many changes associated with implementing the new systems, including timely and effective training, prepared their customers for any changes associated with the new systems, and achieved a reasonable balance in the timing of completion of implementation activities. Although these companies took longer to Go Live and spent more money than initially planned, they successfully avoided the major pitfalls that have rendered so many of these projects less than fully successful. Avista counts its Project Compass in this latter class of successful projects, and is confident in the successful completion of the Project.