DEVELOPMENT OF PRE-PROCESSING AND POST-PROCESSING PROGRAMS FOR NUMERICAL HYDRAULIC ANALYSIS OF IRRIGATION PIPELINES

YOSHIKAZU TANAKA

NARO Institute for Rural Engineering
E-mail: yokka@affrc.go.jp

Abstract- The author proposes a pre-processing and post-processing method using a simplified geographical information system to make the numerical hydraulic analysis of irrigation pipelines easy. In the pre-processing, the input data necessary for analysis can be created easily by processing KML files. These input/output format files in Google Earth make a placemark on Google Earth map. In the post-processing, detailed hydraulic phenomena of pipelines can be displayed visually on Google Earth bird's eye view by creating semi-automatic graph images and 3D graphs from the numerical analysis program output results. The smooth execution of a series of steps for numerical analysis can assist engineers to more efficiently perform designs.

Index Terms- Irrigation pipeline, design, Google Earth , pre-post-processing

I. INTRODUCTION

The total length of irrigation pipelines laid by national and prefectural development programs has reached about 7,500 km (as of June 2013) in Japan. When engineers calculate the design water pressure to determine the pressure limits of a pipe, water hammer pressure must be calculated by executing the numerical water hammer analysis. Engineers require a program that ensures accurate and reliable calculations to support their work.
Irrigation pipeline systems consist of many kinds of incidental and pipe facilities. Generally, small hydraulic units consisting of facilities are designed during the first steps. Later, large hydraulic units combined with small ones are designed. Once a numerical analysis program has been run, computers will execute a numerical hydraulic analysis giving an approximate solution for the water pressure and velocity in a pipe. However, the data input work still imposes a great burden on engineers. Since there are various types of incidental facilities for an irrigation pipeline, the data input work is complicated and time-consuming. The author, an engineer, has proposed a data management technique that enables efficient numerical hydraulic analysis.

It is also important to indicate numerical analysis results clearly for better understanding. During design, an engineer will discuss the system functions of a pipeline under various natural and social conditions for a given area. Therefore, post-processing is necessary to illustrate the entire pipeline system and its details visually on a map.

This paper is intended for engineers performing numerical hydraulic analysis of a given irrigation pipeline. The purpose of this paper is to develop a pre-processing program for easy data input and post-processing to clearly assist visualization of numerical analysis results for better understanding to support engineers.

II. POINTS OF PROBLEM

Problems concerning pre-processing and post-processing are summarized below.

A. Problems concerning Pre-processing

A.1. Case of general numerical analysis programs
General numerical analysis programs, excluding some commercial programs, may not have pre-processing functions.
Therefore, users need to accurately describe files by strictly following the input rules of such programs (Fig. 1). If a hydraulic unit is widely extended or has complicated piping, data input and file management may become difficult.

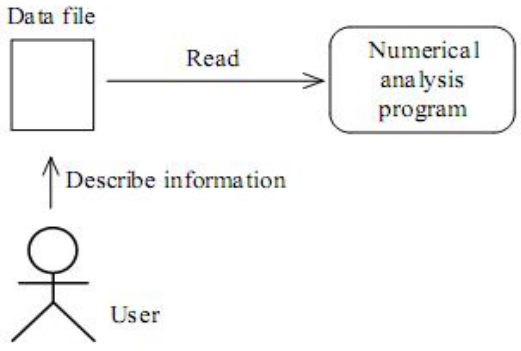

Fig. 1. Conceptual diagram of data input for general numerical analysis program

A.2 Proposed data management

The data management technique proposed in previous reports [1]-[2] defines information

concerning the concept of pipeline in advance as rules and is called the Irrigation System Markup Language (ISML). A system for generating instances from XML files of information about pipe and incidental facilities was created on the basis of rules to handle the same information and to enter initial and boundary conditions data required by the numerical analysis program (Fig. 2). Where, XML stands for eXtensible Markup Language. This is a data expression technology that provides a standard data description format. In XML, input information is arranged by using keyword elements enclosed with <> (hereinafter, referred to as "tags"). ISML is based on XML specifications. Tags such as those shown in Table 1 are used to arrange information in descriptions. The low-order tag configuration depends on each kind of facility. An instance is an entity generated in a computer memory. This is based on the data type called class that is defined in an object oriented language. An object oriented program proceeds with processing by changing the status of an instance value. This technique has the following three advantages:

(1) When designing the hydraulic unit combined with several small hydraulic units to reconsider functions and review specifications, data input can be done more efficiently because the files of small hydraulic units created for individual designs can be reused.

Table 1. Contents of Tags for Facility A

Tag	Meaning	Content of Low-order Tag
<ConnectionInfoOfA>	Connection information	IDs of facilities connected to the upstream and downstream sides
<PositionInfoOfA>	Position information	3D location information of facility.
<StructureInfoOfA>	Structure information	Representative specifications or drawing of facility
<CharacterInfoOfA>	Characteristic information	Value indicating such property as flow velocity coefficient or friction coefficient

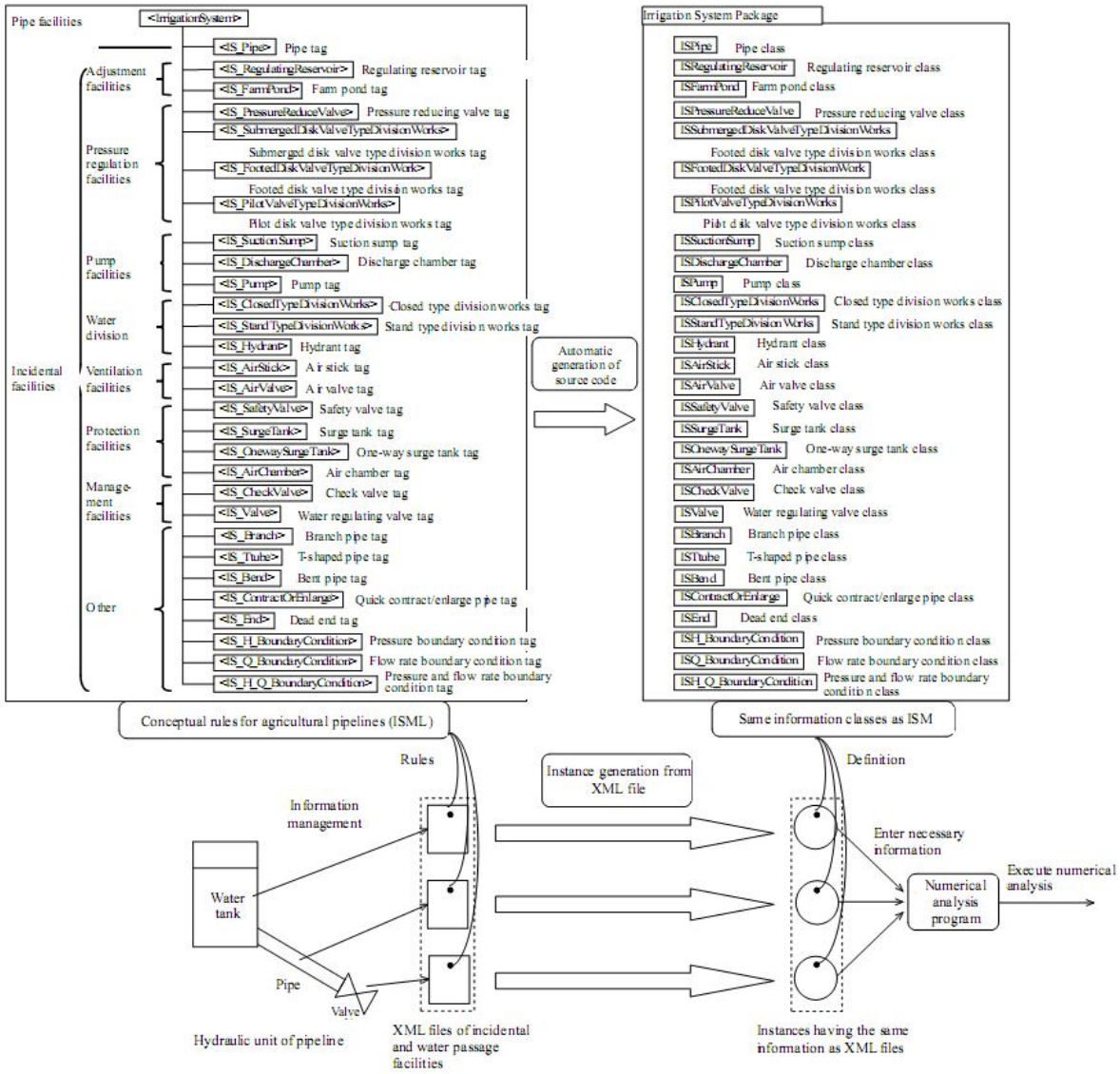

Fig. 2. Conceptual diagram of XML file data management technique

(2) Facility information is arranged with tags and the definitions of input data become clear.

(3) Data input errors can be prevented because the XML parser verifies information described as specified during XM file reading.

The problem with this data management technique is that labor-saving could not be achieved, when engineers create new input files. In addition knowledge about ISML rules and XML are necessary for XML file description.

B. Problems in Post-processing

General numerical analysis programs, excluding some commercial programs, often output numerical analysis results in a CSV format file. The CSV format is describes information by delimiting with a comma. Using spreadsheet software and creating a graph allows an engineer to visually understand the time variations of pressure head and mean flow velocity in pipes (Fig. 3). However, the point where numerical analysis results should be positioned on a map come from the ID or facility name that the engineers decided when creating the input data. The problem with creating input data is that it requires labor and careful attention for a widely extended or complicated hydraulic unit.

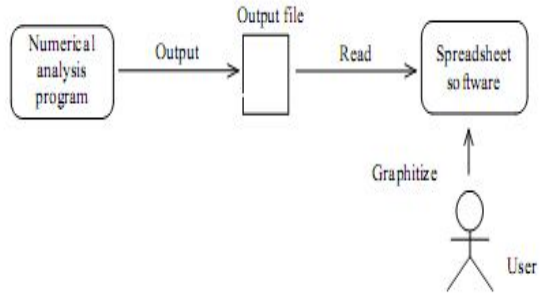

Fig. 3. Conceptual diagram of post-processing for general numerical analysis program

III. PROPOSAL OF PRE-PROCESSING AND POST-PROCESSING

The author proposes a technique for pre-processing and post-processing using a simplified geographical information system (hereinafter, referred to as "GIS") that simplifies data input and visualization of the numerical analysis results. Google Earth is used in this paper. Google Earth refers to dots, lines and other graphics as place marks. Because information can be input and output through KML files, the proposed technique displays results on Google Earth for the pre-processing and post-processing method.

A. About KML

KML is a 3D map information description language specified to display geographical information on Google Earth. In 2008, this was instituted as the standard for the Open Geospatial Consortium (OGC) for promoting GIS standardization. This standard is based on the XML schema language called XML Schema.

Fig. 4 shows the basic configuration of the KML tags. Since not all the tags are necessary, refer to the KML Reference (Google, 2015) for details. Place mark tags are mainly necessary for the technique in this paper. Tags are used to provide place mark information. The description tags under Place mark tags are used to describe place marks (Table 2).

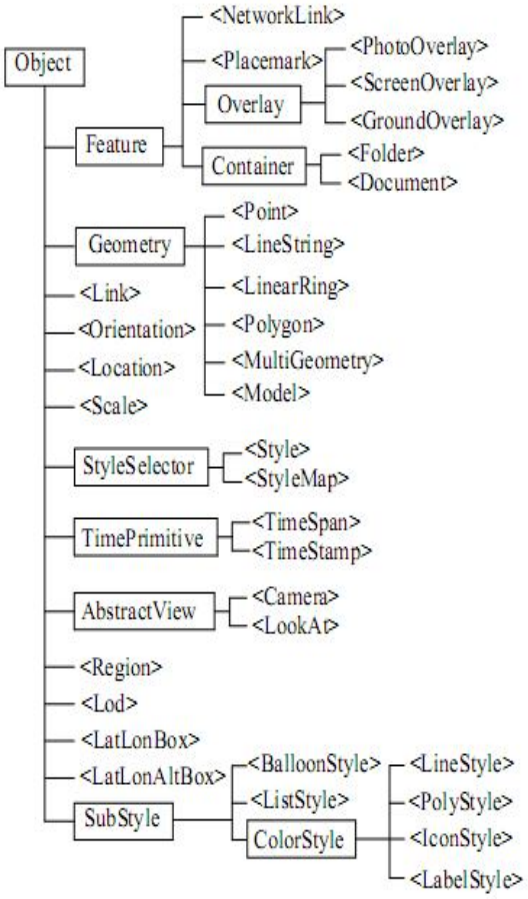

Fig. 4. Basic configuration of KML tags (Source: KML Reference (Google, 2010))

B. About Google Earth

Google Earth is a simplified 3D GIS that displays satellite and aeronautic photos. It is generally used to enter information about specific locations or areas on a map or superpose them on Google Earth similar to other commercial GIS programs. The main features include user-friendliness and the I/O format is KML. For example, Google Earth supports rotation and move commands using a mouse and also allows map viewing. Fig. 5 shows a screen snapshot when a place mark is created on a Google Earth map. In the dialog box for place mark information input, a text area (hereinafter, referred to as "description text area") is displayed to enter a description. Information entered here is described as a character string in the description tag under < Placemark> tag in the KML file.

Table 2. Configuration of tags under <Placemark>

Higher Tag	Lower Tag	Contents
<name>	None	Name of placemark location
<LookAt> or <Camera>	<longitude>	Longitude from the user's point of view
	<latitude>	Latitude from the user's point of view
	<altitude>	Altitude from the user's point of view
	<range>	Range from the user's point of view
	<tilt>	Tilt from the user's point of view
	<heading>	Heading from the user's point of view
	<altitudeMode>	Reference altitude from the user's point of view
<styleUrl>	None	Image of placemark icon
<Point>	<altitudeMode>	Reference altitude of placemark
	<coordinates>	Longitude, latitude, and altitude of placemark point in this order
<description>	None	Description of placemark (A description in the HTML format will be displayed by parsing the HTML syntax.)

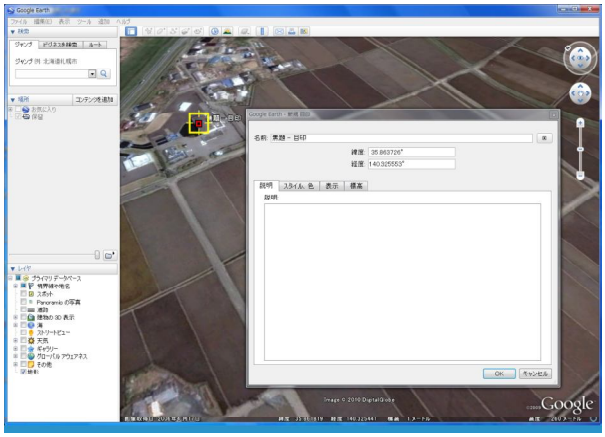

Fig. 5. Example of placemark creation on Google Earth (Google Inc.)

C. Proposed Pre-processing

C.1. Outline

The place marks describe a dot considered as a incidental facility. By creating place marks on a Google Earth map, a group of input files are created. These files include data regarding incidental and pipe facilities based on the first set of input files. The description text for incidental and pipe facilities consists of ISML tags and default values suitable for the respective facilities were pasted in advance to helping users enter information descriptions. A user can create a group of input files for respective facilities by modifying the default values. After users perform an input on Google Earth, the pre-processing program extracts information from the description text areas and creates the XML files for incidental and pipe facilities. Fig. 6 shows a rough flow chart of this procedure. To implement this procedure, in advance, the conventional KML rules defined in the XML Schema must be redefined in RELAX. This enables an automatic generation of a source code concerning the Java class for Relaxer to handle the same information as a KML file. The pre-processing program reads a KML instance generated from this source code and outputs a processed KML file. RELAX stands for Regular Language description for XML. This is a simplified schema language of XML for structured documents based on the Theory of Hedge Automata [3]. RELAX became standard under the Japanese Industrial Standards in 2000. Relaxer is a program capable of generating class source codes of the object oriented language Java to handle the same information as tags based on RELAX rules [4].

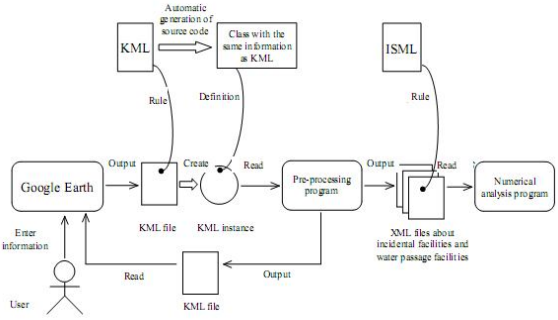

Fig. 6. Conceptual diagram of proposed pre-processing flow

C.2 Pre-processing procedure

Information input and data correction are repeated between Google Earth and the pre-processing program to execute a total of nine steps while processing KML files. Processing can be executed by pressing the right-side conversion button at each step, as can be seen on the pre-processing program screen in Fig. 7. Pre-processing is divided into nine steps because information from the file output of the previous step is used for processing. Dividing the processing work benefits development by making it easy to find and correct errors. The pre-processing work consists of Steps 1 through 3 where input data about incidental facilities are creadted and Steps 4 through 8 where input data about pipe facilities are created.

The procedure is as follows;

Step 1: The user creates dot place marks in Google Earth. The place marks are named after the types of incidental facilities defined in ISML. After creating place marks for all incidental facilities in the hydraulic unit, the user saves the KML file. The pre-processing program reads this KML file, judges the type of incidental facilities from the place mark names, and outputs a group of XML files. This gives the appropriate low-order tags specified in ISML and their appropriate default values.

Step 2: The pre-processing program appends the serial numbers of the incidental facilities from upstream to the place mark names as IDs. The program pastes the content of XML files from Step 1 to <description> tags of the place marks as character strings and then outputs the KML file.

Step 3: The incidental facilities were connected only sequentially from upstream at Step 2. Therefore, the user corrects the connections of the incidental facilities to the appropriate ones in the description text area of each place mark in Google Earth as well as other information about the incidental facilities. After corrections, the KML file is saved.
Step 4: From the KML file, where the connections were corrected, the pre-processing program cuts off character strings in <description> tags of <placemark> indicating the incidental facilities. Then the program again outputs the XML files. The XML files of incidental facilities output at Step 1 are overwritten to complete the final input files.
Step 5: A group of input files about pipe facilities are created. The pre-processing program analyzes the connections of the KML file created in Step 3 and outputs a tentative group of XML files about pipe facilities.
Step 6: The pre-processing program calculates the lengths of pipe facilities connecting the incidental facilities and also the numbers of computational grids.

The group of pipe facility XML files with these values added is pasted to <description> tags of <placemark> in the KML file as character strings.
Step 7: The user corrects information about the pipe facilities in Google Earth and saves the KML file. Afterwards the connections can no longer be corrected.
Step 8: From the KML file created in Step 7, the pre-processing program cuts off character strings in <description> tags of <placemark> indicating the pipe facilities. The program again outputs the pipe facility XML files. These XML files in Step 5 are overwritten to complete the final input files about the pipe facilities.
Step 9: A file specifying the hydraulic unit range is output. This assists the numerical analysis program to find the necessary input XML files from the directory.

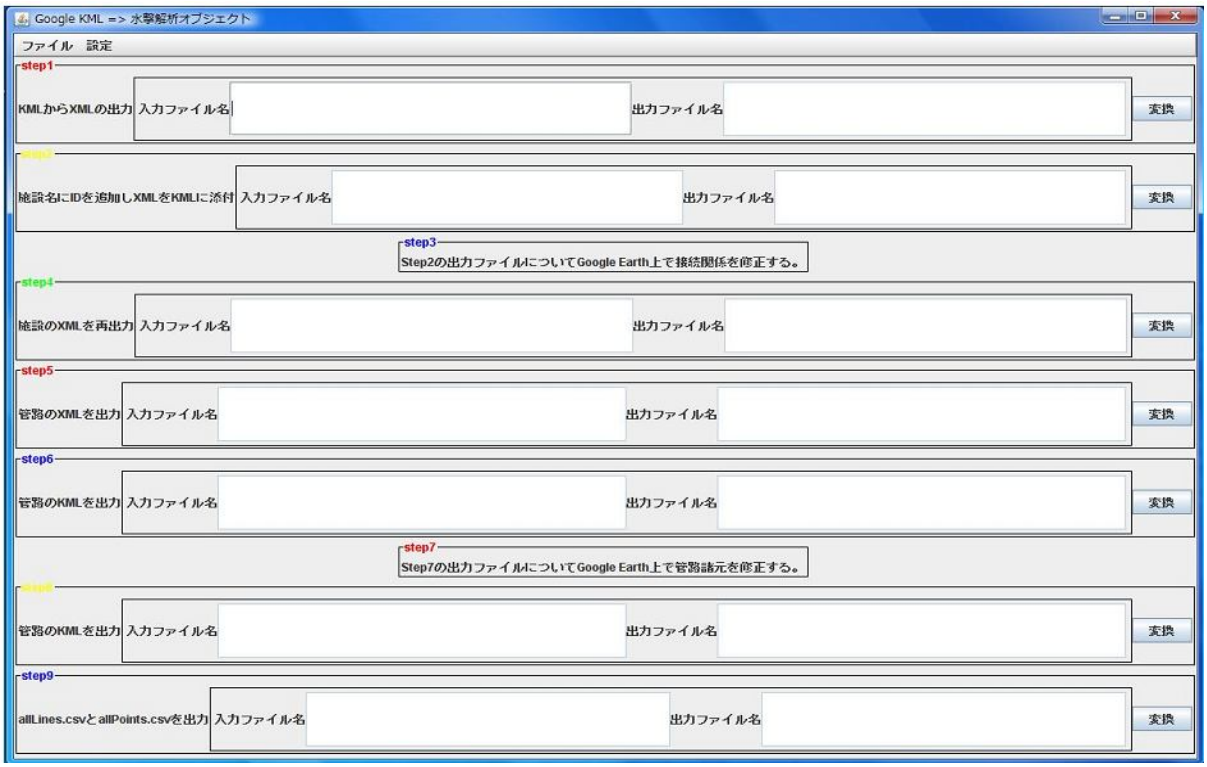

Fig. 7. Windows screen of pre-processing program

C.3. Input method for numerical analysis program

The pre-processing program produces three types of input data files: (1) XML files about incidental and pipe facilities, (2) text files specifying a range of hydraulic units, and (3) text files describing the necessary information for the numerical calculation conditions. The numerical analysis program for water hammer can read files (2) and (3) in the text format for the input, but require a XML parser for type (1) files. The XML parser is prepared in the major programming languages and generates Document Object Model (hereinafter, referred to as "DOM") from the XML files. By reading the DOM, information about incidental and pipe facilities can be entered into the numerical analysis program

D. Proposed Post-processing

D.1. Outline

A CSV file from the numerical analysis program is converted into a KML file and the results are displayed on Google Earth (Fig. 8). KML uses a <Polygon> tag to specify a polygonal shape and a display method for drawing on a map (Table 3). A

bar graph is created with the altitude in the coordinate tag at the bottom of <Polygon>, tag for the maximum water hammer pressure. The <description> tag under

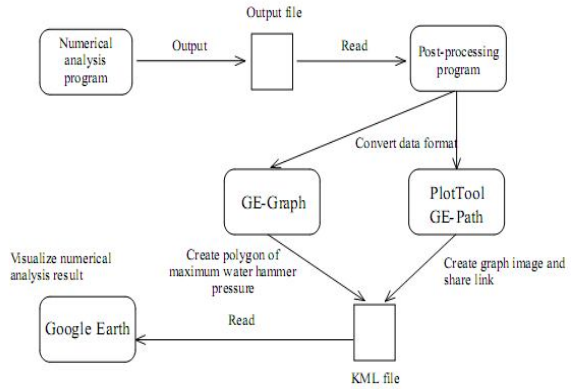

Fig. 8. Conceptual diagram of proposed post-processing

Table 3. Configuration of tags in <Polygon> tag

High-order Tag	Middle Tag	Low-order Tag	Description
<extrude>	None	None	Specify connection between polygon and ground.
<altitudeMode>	None	None	Select an altitude specification method.
<outerBoundaryIs>	<LinearRing>	<coordinates>	Specify four or more points indicating longitude, latitude, and altitude.

a Place mark tag displays the description in a HTML format (Table 2). A hyperlink is shared and a graph image of time-series variation with pressure head is displayed at a place mark for incidental facilities. A program was created to graphitize numerical analysis results and convert them into image files.

The visual file generation program is hereinafter referred to as PlotTool.

Freeware programs called GE-Graph and GE-Path were used for their conversion into a KML file (Ricardo, 2015).

D.2. Post-processing procedure

On the screen of the post-processing program shown in Fig. 9, the Convert button at the right of each step is pressed to execute processing. The procedure is as follows:

Step 1: Output files from the numerical analysis program are converted into the PlotTool input file data format.

Step 2: PlotTool is used to semi-automatically generate graphs of numerical analysis results for all incidental facilities in the hydraulic unit, which are converted into image files.

Step 3: The post-processing program converts data into data formats for GE-Graph and GE-Path and then starts the programs.



Fig. 9. Windows screen of post-processing program

IV. EXAMPLE OF NUMERICAL ANALYSIS

Using an actual pipeline as an analysis example, this chapter verifies the practicality of the developed pre-processing and post-processing programs.

A. Example of a Multi-pipeline System

A hydraulic unit of the trunk pipeline for area N in Prefecture T of Japan was used. This hydraulic unit uses pipe of the pump pressure feed type as shown in Fig. 10. The pipe length is approximatle 6 km and the water level difference between the pump and the discharge chamber 30 m. There are 74 incidental

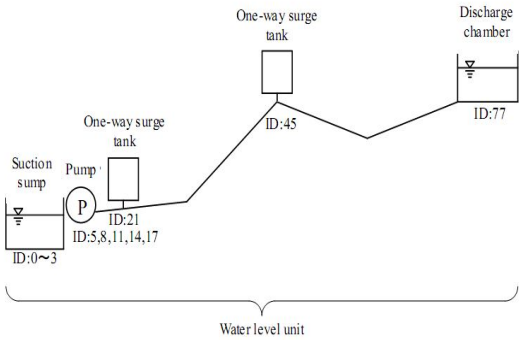

Fig. 10. Basic structure of example hydraulic unit for analysis

facilities between the pump and chamber. Table 4 shows the specifications of five pumps and two surge tanks. By numerical analysis, we reproduced the transient hydraulic phenomenon that would occur if a total of two pumps, large and small, suddenly stopped (400 s after the start of calculation) due to a loss of power.

Instead of simulating design work and entering data based on design drawings, output drawings were used.

Table 4. Pump specifications

Pump Type	Pump Discharge (m^3/min)	Total Head (m)	Quantity
Large	64.5	69	2
Small	33.0	70	2
Retention	12.0	45	1

B. Pre-processing Work Status

By referring to existing drawings, place marks at the locations of incidental facilities were initially created with the types of incidental facilities and their names. At Step 3, we entered XML-format information of the incidental facilities pasted to the text areas for place mark description by correcting the specifications and connections of the facilities with the output drawings as references (Fig. 11). At Step 7, the specifications for pipe facilities with the output drawings were corrected and entered as references because the XML file templates were pasted to the text area for place mark descriptions. Therefore, it is easier to create an input file that complies with the ISML rules.

C. Numerical Analysis Technique

The numerical analysis program is outlined below.

C.1. Basic equations

The basic equations of hydraulic phenomenon in pipes are as follows:(1)

$$g\frac{\partial H}{\partial x}+\frac{\partial V}{\partial t}+V\frac{\partial V}{\partial x}+\frac{fV^2}{2\Phi}=0 \quad (1)$$

$$\frac{a^2}{g}\frac{\partial V}{\partial x}+\frac{\partial H}{\partial t}+V\left(\frac{\partial H}{\partial x}+\sin\gamma\right)=0 \quad (2)$$

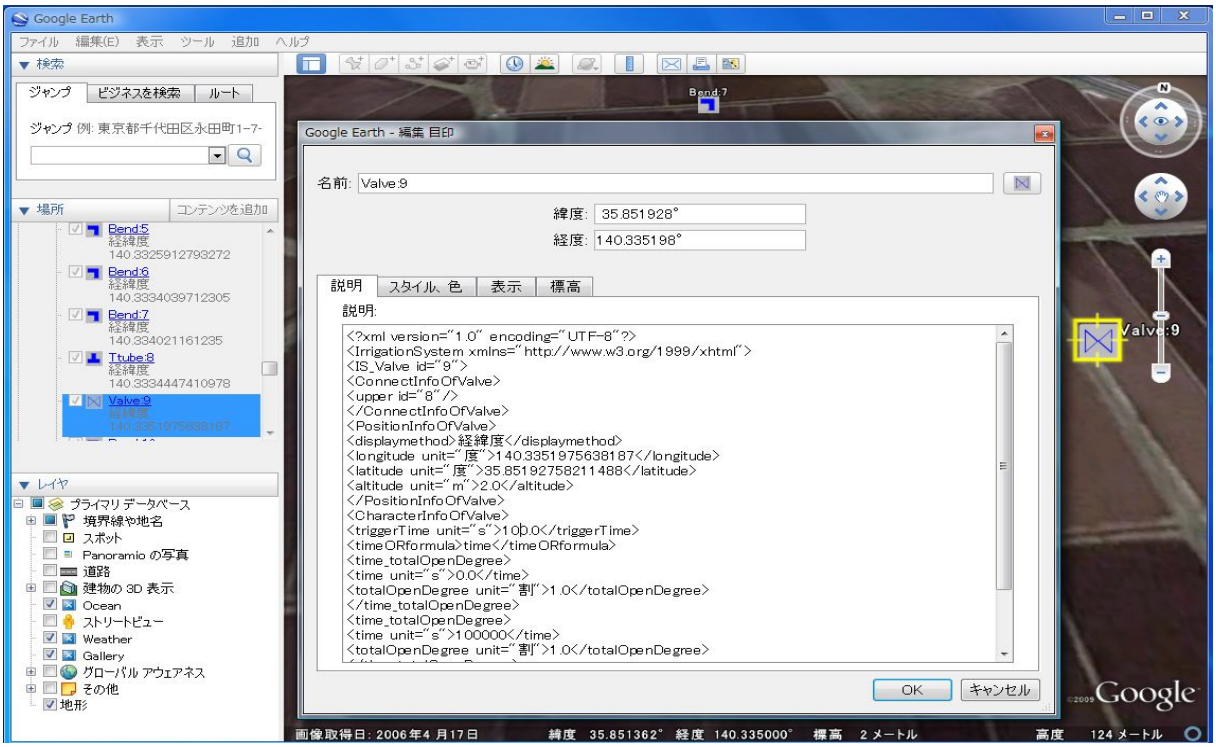

Fig. 11. Information input screen for incidental facilities (using Google Earth)

Where, H: Piezometric head, V: Mean pipe flow velocity, x: Pipe length, t: Time elapsed, G: Gravitational acceleration, Φ: Pipe diameter, a: Pipe pressure propagation velocity, and γ: Pipe gradient

C.2. Method of Characteristics and spatial interpolation

The method of characteristics is used to solve basic equations (3) and (4). Fig. 12 shows the mean pipe flow velocity V_M^{n+1} and pressure head H_M^{n+1} at the next time step n+1 calculated by superposing their values at two places where the two characteristic lines C^+ and C^- intersect with the computational grids of the previous step. Since the characteristic line does not pass computational grid points in most cases, the value at M must be calculated from the values of R and S from equations (5) and (6).

$$H_M^{n+1}=\frac{a}{2g}\left[H_R^n+H_S^n+\frac{a}{g}\left(V_R^n-V_S^n\right)-\frac{f\Delta t}{2\Phi}\left(V_R^n\left|V_R^n\right|-V_S^n\left|V_S^n\right|\right)-\frac{g\left(V_R^n+V_S^n\right)}{a}\sin\gamma\right] \quad (3)$$

$$V_M^{n+1} = \frac{1}{2}[V_R^n + V_S^n + \frac{g}{a}\left(H_R^n - H_S^n\right) - \frac{f\Delta t}{2\Phi}\left(V_R^n\left|V_R^n\right| + V_S^n\left|V_S^n\right|\right) - \frac{g\left(V_R^n - V_S^n\right)}{a}\sin\gamma] \quad (4)$$

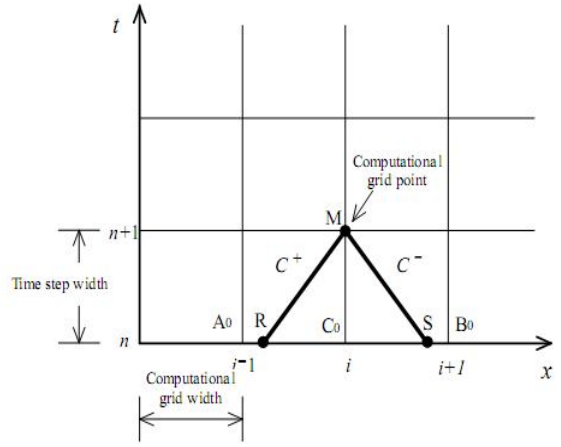

Fig. 12. Positions of characteristic lines and computational grid points

Therefore, the mean pipe flow velocity and pressure head at R and S are spatially interpolated in the sections of A_0-C_0 and B_0-C_0 on the computational grids at the previous step n.

C.3. Execution of numerical analysis

The numerical analysis program screen (Fig. 13) displays the XML files of incidental and pipe facilities created by pre-processing on the second and third lists from the left. Files specify the range of the hydraulic unit displayed on the left-end list with files giving other information necessary for numerical calculations (e.g. time step width, calculation time, range of output, and selection of spatial interpolation). Right-end list gives numerical analysis results after the "Execute" command is selected from the menu. Clicking a file name on a list activates the editor or spreadsheet software for editing.

D. Post-processing Work Status

Numerical analysis results were first converted into a data format for input to PlotTool. Converted files were then selected from the PlotTool screen shown in Fig. 14 and the Draw Graph button was pressed to create a graph of time-series variation with pressure head at all incidental facilities in the hydraulic unit. The Output File button was pressed to convert the graphs into JPEG image files. Image files were created semi-automatically this way. The button at Step 3 was pressed next to convert the files into GE-Graph and GE-Path compatible input formats. Consequently, we can see the entire pipeline on a bird's eye view of Google Earth and determine the maximum water hammer pressure. Clicking a place mark for incidental facilities displays a graph of time-series variation with pressure head at the position. Therefore, the transient hydraulic phenomenon at respective incidental facilities can be verified in detail.

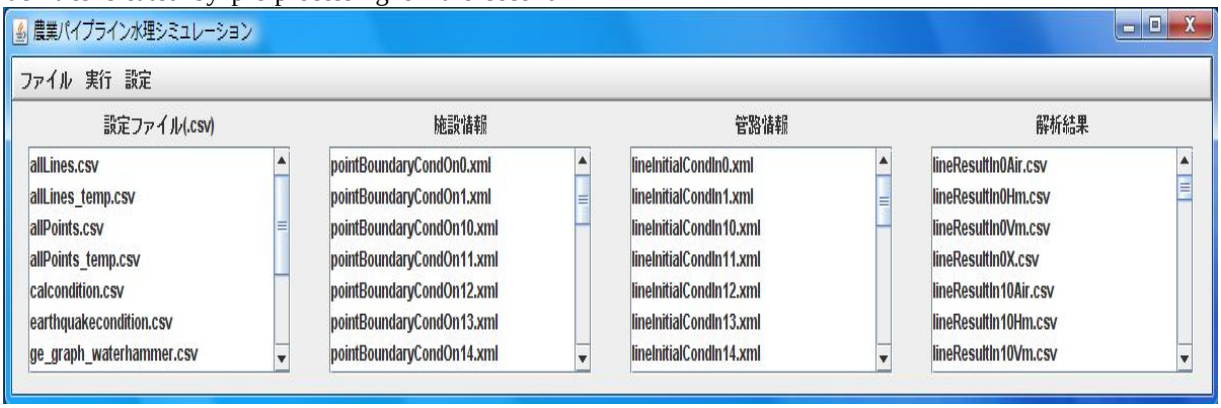

Fig.13 Windows screen of numerical analysis program

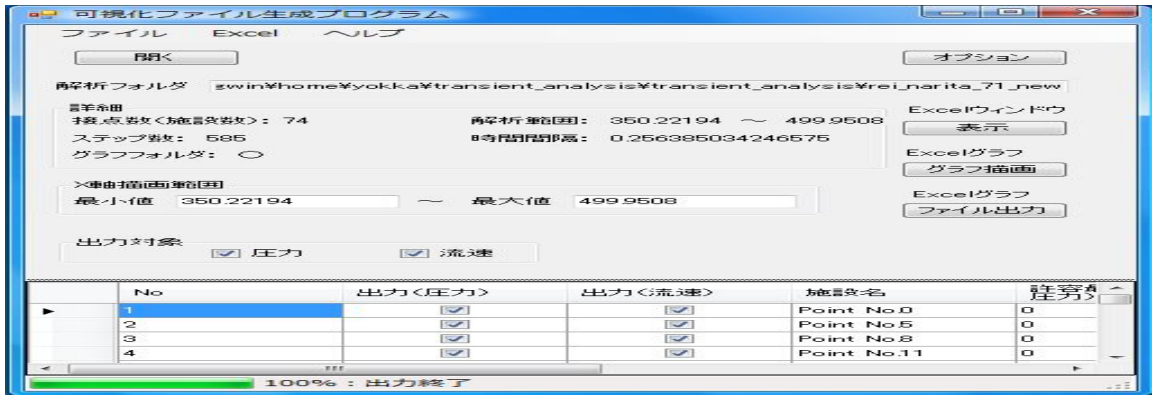

Fig. 14. Windows screen of PlotTool

E. Comparison of Existing Water Hammer Analysis Programs with Work Processes

The practicality of two existing water hammer analysis programs was compared. Here, they are called program A (Japan Association of Agricultural Engineering Enterprises, 2002) and program B (Fuji Research Institute Corporation, 1995). For pre-processing and post-processing, programs A and B both require user work as illustrated in Fig. 1 and Fig. 3. The figures look simple because no program support is provided, but the volume of user work is substantially large. Fig. 15 compares the work

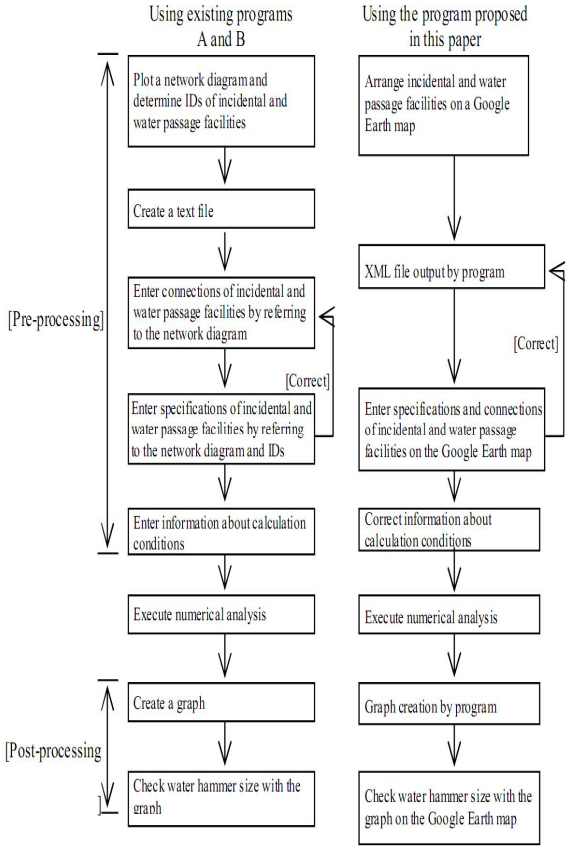

Fig. 15. Comparison of work processes

processes of programs A, B, and the program proposed in this paper. The existing programs A and B draw a network diagram in advance with ID numbers assigned to incidental and pipe facilities from output drawings. By referring to this diagram, the ID numbers and specifications of the respective incidental and pipe facilities are entered into a text file to create an input data file. Meanwhile, the pre-processing program proposed in this paper, automatically assigns serial numbers from upstream as ID numbers if place marks are created for incidental facilities on a Google Earth map. The same processing applies to the ID numbers of pipe facilities. This is expected to reduce errors in writing ID numbers, burden of input and making connections.

DISCUSSION

The pre-processing and post-processing proposed in this paper simplify data input using GIS, as well as the display of numerical analysis results. Therefore, a series of tasks can be performed smoothly for the numerical analysis on a wide range of hydraulic units with large information volumes (Fig. 16).

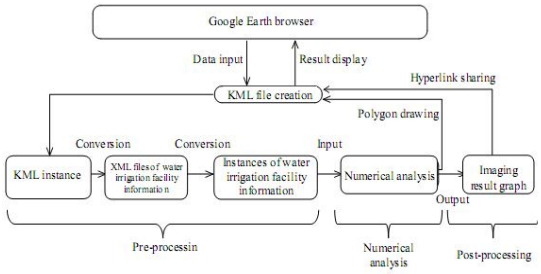

Fig. 16. Numerical analysis workflow

Regarding compatibility and cost efficiency, this technique is suitable because standards are used for XML specifications and other rules. KML is expected to also be applicable to other GIS programs in the future. This technique will enable file transfer based on standards and data reuse for various reviews if applied to program input and output that supports not only the new design, but also functional diagnosis and design updates at the operation management and update stages. Commercial numerical analysis and GIS programs require implementation costs but this technique uses only open sources and freeware. However, the terms of user agreements for software use must be followed.

CONCLUSION

We developed pre-processing and post-processing programs to create files based on ISML and KML for numerical hydraulic analysis of irrigation pipelines. ISML is the standard for our proposed data management technique and KML is the standard for Google Earth. Consequently, the following results were obtained:

(1) Data creation work for numerical hydraulic analysis on pipelines can be reduced by using the Google Earth map display and input functions.
(2) A series of tasks, such as data input, numerical analysis and result visualization can be executed smoothly for a wide range of pipeline scenarios with large information volumes.
(3) Data input work can be simplified even for complicated pipeline systems that are difficult to enter with text file descriptions.
(4) The post-processing program can create an easy to visualize result for pipeline numerical analysis as a bird's eye view on a map

REFERENCES

[1] Y. Tanaka, "The development of water hammer analysis program based on the object oriented development process", Trans. of JDSIDRE, 284, 1-12, 2013

[2] Y.Tanaka, "Development of data management method using XML for hydraulic numerical analysis of the irrigation pipeline", Trans. of JDSIDRE, 284, 13-22, 2013

[3] M. Murata, "XML［I］-XML Schema and RELAX-", The journal of institute of electronics information and communication engineers, 84, 12, 890-894, 2001

[4] T.Asami, "Web development using java/ XML, peason education", 3-108, 2001, (in Japanese)

★★★