WHAT IS CLAIMED IS:

1. A data interpretation apparatus comprising a platform configured to automatically execute type conversion of ~~an~~ objects,

wherein the platform is provided in a control unit of the data interpretation apparatus, and includes an obtaining unit and an interpretation unit, and

the control unit functions such that:

the obtaining unit obtains a plurality of input data as the objects of the platform, the plurality of input data including input data in different representation formats; and

the interpretation unit generates an initial graph with respect to the objects, and performs interpretation by automatically growing a graph from the initial graph ~~while~~ by executing,

downcasting of one of the objects based on a hyponymy-and-hypernymy relationship between the objects, and adding, to the graph, a node associated with the one of the objects for which the downcasting has been executed, the downcasting being a type conversion to a hyponymy-side, and

upcasting of one of the objects based on the hyponymy-and-hypernymy relationship between the objects, and adding, to the graph, a node associated with the one of the objects for which the upcasting has been executed, the upcasting being a type conversion to a hypernymy-side, the downcasting and the upcasting being executed along a conversion path defined such that a meaning and a content of the one of the objects being downcast or upcast does not

changeas necessary, the type conversion of the object associated with each node of the graph toward hyponymy or toward hypernymy.

2. The data interpretation apparatus according to claim 1, wherein the interpretation unit automatically grows the graph by executing downcasting of the object that is a type conversion to the hyponymy-side, based on a hyponymy and hypernymy relationship between objects, and adding, to the graph, a node associated with an object that has been downcast.

32. A data interpretation method using a platform configured to automatically execute type conversion of an objects,

wherein the platform is provided in a control unit of a computer, and includes an obtaining unit and an interpretation unit, and

wherein the obtaining unit executes a step of obtaining a plurality of input data as the objects of the platform, the plurality of input data including input data in different representation formats; and

the interpretation unit executes a step of generating an initial graph with respect to the objects, and performing interpretation by automatically growing a graph from the initial graph while by executing,

downcasting of one of the objects based on a hyponymy-and-hypernymy relationship between the objects, and adding, to the graph, a node associated with the one of the objects for which the downcasting

has been executed, the downcasting being a type conversion to a hyponymy-side, and

upcasting of one of the objects based on the hyponymy-and-hypernymy relationship between the objects, and adding, to the graph, a node associated with the one of the objects for which the upcasting has been executed, the upcasting being a type conversion to a hypernymy-side, the downcasting and the upcasting being executed along a conversion path defined such that a meaning and a content of the one of the objects being downcast or upcast does not changeas necessary, the type conversion of the object associated with each node of the graph toward hyponymy or toward hypernymy.

43. A program configured to be executed by a computer,

wherein in a platform configured to automatically execute type conversion of an objects, the platform being provided in a control unit of the computer, and including an obtaining unit and an interpretation unit, the program causes the control unit to operate such that:

the obtaining unit executes a step of obtaining a plurality of input data as the objects of the platform, the plurality of input data including input data in different representation formats; and

the interpretation unit executes a step of generating an initial graph with respect to the objects, and performing interpretation by automatically growing a graph from the initial graph while by executing,

downcasting of one of the objects based on a

hyponymy-and-hypernymy relationship between the objects, and adding, to the graph, a node associated with the one of the objects for which the downcasting has been executed, the downcasting being a type conversion to a hyponymy-side, and

upcasting of one of the objects based on the hyponymy-and-hypernymy relationship between the objects, and adding, to the graph, a node associated with the one of the objects for which the upcasting has been executed, the upcasting being a type conversion to a hypernymy-side, the downcasting and the upcasting being executed along a conversion path defined such that a meaning and a content of the one of the objects being downcast or upcast does not changeas necessary, the type conversion of the object associated with each node of the graph toward hyponymy or toward hypernymy.

54. A data integration apparatus comprising a platform configured to automatically execute type conversion of an objects,

wherein the platform is provided in a control unit, and includes an obtaining unit and an interpretation unit, and

the control unit functions such that:

the obtaining unit obtains a plurality of input data as the objects of the platform, the plurality of input data including input data in different representation formats; and

the interpretation unit generates an initial graph with respect to the objects, performs interpretation by automatically growing a graph from the initial graph while by executing,

downcasting of one of the objects based on a hyponymy-and-hypernymy relationship between the objects, and adding, to the graph, a node associated with the one of the objects for which the downcasting has been executed, the downcasting being a type conversion to a hyponymy-side, and

upcasting of one of the objects based on the hyponymy-and-hypernymy relationship between the objects, and adding, to the graph, a node associated with the one of the objects for which the upcasting has been executed, the upcasting being a type conversion to a hypernymy-side, the downcasting and the upcasting being executed along a conversion path defined such that a meaning and a content of the one of the objects being downcast or upcast does not changeas necessary, the type conversion of the object associated with each node of the graph toward hyponymy or toward hypernymy, and establishes integrated data obtained by integrating one or more objects.

6. A data integration method using a platform configured to automatically execute type conversion of an objects,

wherein the platform is provided in a control unit of a computer, and includes an obtaining unit and an interpretation unit, and

wherein the obtaining unit executes a step of obtaining a plurality of input data as the objects of the platform, the plurality of input data including input data in different representation formats; and

the interpretation unit executes a step of

generating an initial graph with respect to the objects, performing interpretation by automatically growing a graph from the initial graph ~~while~~ by executing,

downcasting of one of the objects based on a hyponymy-and-hypernymy relationship between the objects, and adding, to the graph, a node associated with the one of the objects for which the downcasting has been executed, the downcasting being a type conversion to a hyponymy-side, and

upcasting of one of the objects based on the hyponymy-and-hypernymy relationship between the objects, and adding, to the graph, a node associated with the one of the objects for which the upcasting has been executed, the upcasting being a type conversion to a hypernymy-side, the downcasting and the upcasting being executed along a conversion path defined such that a meaning and a content of the one of the objects being downcast or upcast does not change~~as necessary, the type conversion of the object associated with each node of the graph toward hyponymy or toward hypernymy, and establishing integrated data obtained by integrating one or more objects~~.

~~7~~6. A program configured to be executed by a computer,

wherein in a platform configured to automatically execute type conversion of ~~an~~ objects, the platform being provided in a control unit of the computer, and including an obtaining unit and an interpretation unit, the program causes the control unit to operate such that:

the obtaining unit executes a step of obtaining a plurality of input data as the objects of the platform, the plurality of input data including input data in different representation formats; and

the interpretation unit executes a step of generating an initial graph with respect to the objects, performing interpretation by automatically growing a graph from the initial graph ~~while~~ by executing,

downcasting of one of the objects based on a hyponymy-and-hypernymy relationship between the objects, and adding, to the graph, a node associated with the one of the objects for which the downcasting has been executed, the downcasting being a type conversion to a hyponymy-side, and

upcasting of one of the objects based on the hyponymy-and-hypernymy relationship between the objects, and adding, to the graph, a node associated with the one of the objects for which the upcasting has been executed, the upcasting being a type conversion to a hypernymy-side, the downcasting and the upcasting being executed along a conversion path defined such that a meaning and a content of the one of the objects being downcast or upcast does not change~~as necessary, the type conversion of the object associated with each node of the graph toward hyponymy or toward hypernymy, and establishing integrated data obtained by integrating one or more objects~~.

~~8~~7. A digital city establishing system comprising:

a data interpretation apparatus comprising a

platform configured to perform automatic conversion of type conversion of objects according to claim 1; and

a plurality of program elements,

wherein the platform achieves loose coupling between the plurality of program elements by abstracting input formats of the plurality of program elements and expanding an application range, through pathfinding and automatic execution of the type conversion based on a hyponymy-and-hypernymy relationship between the objects.