1. Introduction

Software requirement specification (SRS) is a complete description of the behavior of a system to be developed. It includes a set of use cases that describe all the interactions the users will have with the software. Use cases are also known as functional requirements. In addition to use cases, the SRS also contains non-functional (or supplementary) requirements. Non-functional requirements are requirements which impose constraints on the design or implementation (such as performance engineering requirements, quality standards, or design constraints). In a few cases, the user and system requirements may be integrated into a single description. In other cases, the user requirements are defined in an introduction to the system.
requirements specification. If there are a large number of requirements, the detailed system requirements may be presented in a separate document.

Specification and design are inextricably intermingled. Architectural design is essential to structure a specification and the specification process. Formal specifications are expressed in a mathematical notation whose vocabulary, syntax and semantics are formally defined [1]. Usually both the system requirements and the system design are expressed in detail and carefully analyzed and checked before implementation begins.

The necessary information content and recommendations for an organization for software design descriptions (SDDs) are described. An SDD is a representation of a software system that is used as a medium for communicating software design information. This recommended practice is applicable to paper documents, automated databases, design description languages, or other means of description. If a formal specification of the software is developed, this usually comes after the system requirements have been specifying but before the detailed system design. There is a tight response loop between the detailed requirements specification and the formal specification as shown in Figure 1 [2]. One of the main benefits of formal specification is its ability to uncover problems and ambiguities in the system requirements.

![Figure 1. The Requirement Specification Process](image)

The principal value of using formal methods in the software process is that it forces an analysis of the system requirements at an early stage. Correcting errors at this stage is cheaper than modifying a delivered system. Formal methods consist of formal specification, specification analysis and proof, transformational development and program verification. Formal specification techniques are most cost-effective in the development of critical systems where safety, reliability and security are particularly important.

Nowadays, formal proving can be done with the support of formal method tools such as theorem proven tools. Theorem proven is a tool that implements automatic theorem proving need of user support. Regularly, developers cover a long times and looping process, so there might be a great possibility of mistakes. The proofs are efficiently when it been presented in a user-friendly approach and it should not be unreasonably large. Nevertheless, a lot of the proof that involved in software validation is naturally detailed, low-level and repetitious.

So we can briefly state that it is unsuitable for human checking. Thus, formal proving supported by tool, which is not only reduce the possibility of mistakes but also not totally removes it [3]. Hence, the use of support tool is a main factor that can affect the acceptance of formal method practically [4].

The Z specification language is a way of decomposing a specification into small pieces called schemas [5]. Each piece can be linked with a commentary that gives explanation informally the importance of the formal mathematics. A schema is essentially the formal specification analogous to programming language subroutines that are used to structure a
system, where the schemas are used to structure a formal specification. The Z is physically powerful on sets and functions. Generally, Z notation is used for sequential situation. We interested in using Z notation because it is a mature technique for model-based specification [6].

2. SIDIF Framework

Spatial information databases integration framework or SIDIF is a design database integration space proposed to evaluate development effectiveness artificial reef. This method was one method regarded as a new idea to ensure effectiveness of artificial reef development project can be valued it effectiveness.

As evaluation process previously require fairly high cost required one special scuba unit to assess artificial reef levels of development and development by diving method. SIDIF was one method for information combination of two or more database which possess different scheme and also same. In this study, research only made integration of two database only namely artificial reef database (ARPOS) and fish landing database (WIFISH) [7], [8]. Before integration process both this database is conducted, various issues should be identified over proceed. A few process or move need to be taken broken up into 4 levels as follows [9]:

i. **Pre-integration**: Process to assess database environment are used example like Oracle, MySql, MSSQL, MS ACESSS and form other database.

ii. **Scheme comparison**: Scheme comparison or structure for each this database is needed to facilitate integration process conducted. In early, this process is made manually.

iii. **Intermediary software development (middleware)**: an application shall be developed for integration process the data base workable.

iv. **Post-integration**: Integration process assessment was being conducted from credibility process aspect and “interoperability”.

The evaluation results of the two databases are found that the location (position) of artificial reef can be equal with the location of fish catches conducted. Due to this, the location based technique is a core and a fundamental to determine the effectiveness development level of the artificial reef project development at a certain location and also within the timeline. An equivalent assumption can be formed as follows:

*Catch location (CL) is equivalent with artificial reefs development (AR). Assessment factor is dependent on catch yield number (CT) for each type of fish (FT) and comparison with artificial reef type (RT) which included. A formula can be set up here. CL ≡ AR where if found CT ≡ FT is high then AR is effective which depending on catch date (DT) made. Vice versa if CT low, then AR to be ineffective.*

Based on that assumption formula, a newly developed algorithm called location based technique can assess the effectiveness level of artificial reef development project. The integration of location based process and scheme comparison would be made by intermediary software (middleware) which shall be developed as shown in Figure 2.

![Figure 2. SIDIF architecture](image-url)
Since those data in the three different databases are at different location and in different formats, then intermediary software (middleware) is very much needed for integration purposes [10]. Table 1 until Table 5 below shows the information about the data that convert to specific database type in different environment [11].

### Table 1. Artificial reefs (AR) distribution information

<table>
<thead>
<tr>
<th>Code</th>
<th>Areas</th>
<th>Latitude</th>
<th>Longitude</th>
<th>Type</th>
<th>Total</th>
<th>Year Built</th>
<th>Total Cost</th>
<th>Granted By</th>
</tr>
</thead>
<tbody>
<tr>
<td>TTS01</td>
<td>Pulau Bidong</td>
<td>1000</td>
<td>123555</td>
<td>KUBOID</td>
<td>950</td>
<td>1998</td>
<td>49970</td>
<td>KWP</td>
</tr>
<tr>
<td>TTS01</td>
<td>Pulau Bidong</td>
<td>1000</td>
<td>123555</td>
<td>KUBOID</td>
<td>950</td>
<td>1998</td>
<td>49970</td>
<td>KWP</td>
</tr>
<tr>
<td>TTS16</td>
<td>Pulau Perhentian</td>
<td>200</td>
<td>205</td>
<td>SERAMIK</td>
<td>180</td>
<td>2003</td>
<td>343300</td>
<td>DANA KHAS</td>
</tr>
<tr>
<td>TTS04</td>
<td>Pulau Kapas</td>
<td>95</td>
<td>100</td>
<td>KUBOID</td>
<td>1400</td>
<td>1999</td>
<td>122900</td>
<td>KWP</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Boat No.</th>
<th>Landing Date (2009)</th>
<th>Landing Time</th>
<th>Fish Code</th>
<th>Fish Type</th>
<th>Total Landing</th>
<th>Area of Fishing</th>
<th>Artificial Reef Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>TAP001</td>
<td>Dec 12</td>
<td>11.00</td>
<td>AA1</td>
<td>Kerisi</td>
<td>100</td>
<td>Pulau Bidong</td>
<td>Kuboid</td>
</tr>
<tr>
<td>TAP001</td>
<td>Dec 12</td>
<td>11.00</td>
<td>AA2</td>
<td>Kembong</td>
<td>200</td>
<td>Pulau Kapas</td>
<td>Kuboid</td>
</tr>
<tr>
<td>TAP002</td>
<td>Dec 13</td>
<td>10.00</td>
<td>AA1</td>
<td>Kerisi</td>
<td>100</td>
<td>Pulau Perhentian</td>
<td>Seramik</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Fisherman Name</th>
<th>Reg. No</th>
<th>Boat No</th>
<th>Loading</th>
<th>Fishing Zone</th>
<th>Main Fishing Equipment</th>
<th>Other Fishing Equipment</th>
<th>License no.</th>
<th>License expiry Date</th>
<th>Crew No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ahmad bin Ali</td>
<td>1112</td>
<td>TAP001</td>
<td>100</td>
<td>Zon A</td>
<td>Bubu</td>
<td></td>
<td>1234</td>
<td>20/2/2012</td>
<td>8</td>
</tr>
<tr>
<td>Ahmad Durah B. Sameon</td>
<td>2222</td>
<td>Tap002</td>
<td>500</td>
<td>Zon B</td>
<td>Tiada</td>
<td></td>
<td>5678</td>
<td>23/3/2013</td>
<td>4</td>
</tr>
</tbody>
</table>

### Table 4. Integration two Tables with New Information (SIDIF)

<table>
<thead>
<tr>
<th>Landing date (2009)</th>
<th>Fish Type</th>
<th>Total Landing</th>
<th>Area of Fishing</th>
<th>Artificial Reef Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dec 12</td>
<td>Kerisi</td>
<td>100</td>
<td>Pulau Bidong</td>
<td>Kuboid</td>
</tr>
<tr>
<td>Dec 12</td>
<td>Kembong</td>
<td>200</td>
<td>Pulau Kapas</td>
<td>Kuboid</td>
</tr>
<tr>
<td>Dec 13</td>
<td>Kerisi</td>
<td>100</td>
<td>Pulau Perhentian</td>
<td>Seramik</td>
</tr>
</tbody>
</table>

### Table 5. Integration of three Tables with New Information (SIDIF)

<table>
<thead>
<tr>
<th>Landing date (2009)</th>
<th>Boat No.</th>
<th>Fish Type</th>
<th>Total Landing</th>
<th>Area of Fishing</th>
<th>Artificial Reef Type</th>
<th>Fishing Zone</th>
<th>Main Fishing Equipment</th>
<th>Other Fishing Equipment</th>
<th>Crew No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dec 12</td>
<td>TAP001</td>
<td>Kerisi</td>
<td>100</td>
<td>Pulau Bidong</td>
<td>Kuboid</td>
<td>Zon A</td>
<td>Bubu</td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>Dec 12</td>
<td>TAP001</td>
<td>Kembong</td>
<td>200</td>
<td>Pulau Kapas</td>
<td>Kuboid</td>
<td>Zon A</td>
<td>Bubu</td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>Dec 13</td>
<td>TAP002</td>
<td>Kerisi</td>
<td>100</td>
<td>Pulau Perhentian</td>
<td>Seramik</td>
<td>Zon B</td>
<td>Tiada</td>
<td></td>
<td>4</td>
</tr>
</tbody>
</table>

### 3. Formal Specification Development using Z-eves Tools

Based on the information in Table 1 until Table 4, the formal specification is generated using Z-eves tools for formalizing the software specification for all the information. Figure 3 until Figure 5 illustrated on the schema for all the information with different type of databases in single server.
**Figure 3. One of Z Schema for Artificial Reefs Distribution Database**

Formal Specification for Spatial Information Databases Integration .... (Mustafa Man)
Figure 4. One of Z Schema for Daily Fish Landing Database
4. Result/Output Validation

Once the formal specification and validation was successful, then coding process is done using .net programming language for developing a middleware application. Below is one module of coding for integrating two databases which is included in SIDIF framework. The output for this module is shown in Figure 6 as below.

```
//combining the data from two different databases
DataTable combine = new DataTable();
combine.Columns.Add("TARIKHTANGKAPAN", typeof(string));
combine.Columns.Add("JENISIKAN", typeof(string));
combine.Columns.Add("JUMLAHHASILTANGKAPAN", typeof(string));
combine.Columns.Add("KAWASANTANGKAPAN", typeof(string));
combine.Columns.Add("JENISTUKUN", typeof(string));

for (int i = 0; i < dt1.Rows.Count; i++)
{
    for (int j = 0; j < dt2.Rows.Count; j++)
    {
        if (dt1.Rows[i]["TAPAKUNJAM"].ToString() == dt2.Rows[j]["KAWASANTANGKAPAN"].ToString())
        {
            combine.Rows.Add(dt2.Rows[j]["TARIKHPENDARATAN"].ToString(),
                dt2.Rows[j]["JENISIKAN"].ToString(),
                dt2.Rows[j]["JUMLAHHASILTANGKAPAN"].ToString(),
                dt1.Rows[i]["TAPAKUNJAM"],
                dt1.Rows[i]["JENISTUKUN"]);
        }
    }
}
GridView3.DataSource = combine;
GridView3.DataBind();
```

Figure 6. Output for databases Integration Using SIDIF
5. Conclusion

SIDIF is an architecture design which can be recycled in integration problem solutions of various databases. Location based technique for determination of the artificial reef is better to obtain satisfactory decision. Due to this, if this method is successfully deployed and implemented, then the determination of effectiveness for artificial reef development project can be valued easily without requiring high cost and can give huge impact to fishing industry in Malaysia.

This research has demonstrated that the validation on Z formal specification to the case study in fishery industry databases can reduce overall development time, can overcome ambiguity by which at the same time allow an early detection of errors (if exist). According to pass experiences, many theorems have been through a long and repetitious proving process. If the proving is done manually by humans, the possibility of mistake made is high. Using Z/EVES, not only this possibility can be reduced, the proving can be done fast and reliable. One of our future works shall deal with complete and precise specification and validation for multiple databases in distributed database systems environment.

References