

<p align="center">&lt;&lt;ICH&gt;&gt; <b>Interface</b></p>
<p>+ field1: Type + field2: Type</p>
<p>+ method1(Type): Type + method2(Type, Type): Type Approx 4,000 fields. Relates to every Class as a Business Channel Owner</p>

<b>&lt;&lt;IREA&gt;&gt;</b> <b>Interface</b>
+ field1: Type + field2: Type
+ method1(Type): Type + method2(Type, Type): Type Fewer than 4,000 fields. Relates to every Class as a Business Channel Owner.

<pre> &lt;&lt;Supplier&gt;&gt; Interface  + field1: Type + field2: Type  + method1(Type): Type + method2(Type, Type): Type Restricted fields (including CCustomer class). Relates to some Classes only, as a Business Channel Owner's Supplier. </pre>
--

Interface
+ field1: Type + field2: Type
+ method1(Type): Type + method2(Type, Type): Type Limited array of fields. Possible to access public sales & conveyancing services, independently. Otherwise relates as a customer
<< Buyer>>

+ field1: Type + field2: Type
+ method1(Type): Type + method2(Type, Type): Type Limited array of fields Possible to access public sales & conveyancing services. Otherwise relates as a customer

<div>&lt;&lt;Contact&gt;&gt;</div> <div>Interface</div>
<div>+ field1: Type</div> <div>+ field2: Type</div>
<div>+ method1(Type): Type</div> <div>+ method2(Type, Type): Type</div> <div>Can be vehicle to enter a contact as a customer via ActiveLink.</div> <div>Property preferences carry across. May also serve as simple contact store. Relates as 3rd party.</div>

<b>&lt;&lt;Landlord&gt;&gt;</b> <b>Interface</b>
+ field1: Type + field2: Type
+ method1(Type): Type + method2(Type, Type): Type
Possible to access public letting, & repair services independently. Otherwise relates as a customer.

<p align="center">&lt;&lt;IApplicant&gt;&gt;</p> <p align="center"><b>Interface</b></p>
<p>+ field1: Type</p> <p>+ field2: Type</p>
<p>+ method1(Type): Type</p> <p>+ method2(Type, Type): Type</p>
<p>Possible to access public leasing &amp; tenanting services independently.</p> <p>Otherwise relates as a customer.</p>

<div> <div>&lt;&lt;Tenant&gt;&gt;</div> <div>Interface</div> </div>
<div> <div>+ field1: Type</div> <div>+ field2: Type</div> </div>
<div> <div>+ method1(Type): Type</div> <div>+ method2(Type, Type): Type</div> </div>
<div> <div>Possible to access public leasing,</div> <div>tenanting &amp; support services</div> <div>independently. Otherwise relates</div> </div>

<b>&lt;&lt;IResident&gt;&gt;</b> <b>Interface</b>
+ field1: Type + field2: Type
+ method1(Type): Type + method2(Type, Type): Type

Possible to access public leasing, residential & support services independently. Otherwise relates

as a customer.
----------------

  

<p align="center">&lt;&lt;IStakeholder&gt;&gt;</p> <p align="center"><b>Interface</b></p>
<p>+ field1: Type</p> <p>+ field2: Type</p>
<p>+ method1 (Type): Type</p> <p>+ method2 (Type, Type): Type</p>

A Stakeholder has access to the appropriate information for the

task at hand. This would not be extensive in coverage of the database. Relates as a 3rd party
<div style="text-align: center;"> <b>&lt;&lt;ISolicitor&gt;&gt;</b>  <b>Interface</b> </div> <div> + field1: Type  + field2: Type    + method1(Type): Type  + method2(Type, Type): Type </div>

Relates to some classes as a Business Channel Owner and others as a Regulator. Fields for Conveyancing.
<b>&lt;&lt;IRegulator&gt;&gt; Interface</b>
+ field1: Type + field2: Type
+ method1(Type): Type + method2(Type, Type): Type A regulator has access to the

CProperty
+ field: type
+ field: type
+ field: type

CCustomer
+ field: type
+ field: type
+ field: type

```
+ field: type
+ field: type
+ field: type
```

```

class CSupplier {
+ field: type
+ field: type
+ field: type
}

class CSchedule {
+ field: type
+ field: type
+ field: type
}

```

```
+ field: type
+ field: type
+ field: type
+ field: type
+ field: type
+ field: type
```

```
+ field: type
+ field: type
+ field: type
+ field: type
+ field: type
+ field: type
```

+ field: type
+ field: type
+ field: type
+ field: type
+ field: type
+ field: type

- + field: type
- + field: type
- + field: type

CRegulator
+ field: type
+ field: type
+ field: type

CRegister
+ field: type
+ field: type
+ field: type

All enums
+ field: type
+ field: type
+ field: type

