

# 실시간 환경 센서 데이터의 3차원 시각화 시스템 설계 및 구현

Design and Implementation of 3D Visualization System for Real-time Environmental Sensor Data

김경옥, 반경진, 류남훈, 김응곤  
순천대학교 컴퓨터과학과

KyeongOg Kim, KyeongJin Ban, NamHoon Ryu,  
EungKon Kim  
Dept. of Computer Science, Sunchon National  
University

## 요약

이전에는 문자기반 인터페이스만으로도 데이터의 분석이 충분했지만 오늘날의 사용자들은 그래픽인터페이스에 익숙해져 있으며, 사용자 인터페이스에 대한 요구사항은 갈수록 증가하고 있는 추세이다.

그러나 사용자들의 다양한 욕구를 충족시킬 수 있는 인터페이스를 디자인하는 일은 소프트웨어 개발자의 기술만으로는 부족하기에 전문 디자이너가 필요하지만 이런 모든 기능을 갖춘 인터페이스를 개발하기 위해 디자이너와 개발자가 공동으로 작업하는 데는 많은 제약이 따른다. 또한 3차원 형태의 그래프 및 애니메이션 기법 등을 활용한 사용자 인터페이스의 개발은 개발비용의 증가를 초래하게 된다.

본 논문에서는 디자이너와 개발자의 공동 작업이 가능하며 2차원 및 3차원 형태의 그래프, 애니메이션, 소리 등 다양한 멀티미디어 기능의 구현이 가능한 WPF (Windows Presentation Foundation)를 활용하여 각종 환경 센서 및 계측장비로부터 얻은 실시간 데이터의 3차원 시각화 시스템을 설계하고 구현하였다.

## Abstract

Although data analysis in earlier days had been sufficiently done only by character user interface, users of these days are more used to the graphic user interface and the requirements for the user interface are gradually varying and increasing. In order to meet users' various wants and needs and to develop well-equipped interface, not only software developers but also professional designers who can complement the technique of the developers are needed. But in reality there are many restrictions and difficulties for developers and designers to work together cooperatively. Moreover, developing user interface in use of 3D type of graphics and animation techniques causes the rise of the developing cost.

The thesis attempts to design and implement 3D visualization for real-time sensor data collected by the various environmental sensor and measuring devices, by using WPF (Window Presentation Foundation) which can make both developers and designers work together cooperatively and which makes it possible to implement various multimedia functions such as a 2D and 3D type of graphics, animation techniques, and an acoustic effect.

## I. 서론

기존에는 정보를 분석함에 있어서 2차원 형태로도 충분했었다[1]. 그러나 오늘날의 사용자들은 컴퓨터의 처리 속도 향상으로 인해 사용자 인터페이스에 대한 요구사항이 갈수록 증가하고 있는 추세이다. 또한 각종 정보에 대해서도 즉각적인 인식이 가능한 화면을 요구하고 있다[2][3].

양식농장은 농장 관리를 위해 다양한 요소들의 모니터링이 필요하다. 환경 요소들의 약간의 변화로도 어류 양식에는 큰 피해가 올 수 있기 때문에 대량의 어류를 양식하는 곳일수록 더욱 더 절실하다. 환경 센서 및 측정 장비의 역할은 농장의 여러 가지 형태의 요소들을 관리하는 것이다. 예를 들면 pH,

용존 산소량, 수위, 수온, 유속 등을 감시하기 위해 필요하다 [4].

이 논문에서는 각종 센서 및 측정 장비로부터 받은 실시간 데이터를 2D 및 3D 그래픽, 애니메이션 형태 등의 다양한 시각화 방법으로 구현하는 방법을 제안한다.

## II. 관련연구

### 1. 닷넷(.NET)

마이크로소프트가 설계·구현한 대표적인 컴포넌트 아키텍처로 COM(Component Object Model)이 있다[5]. COM은

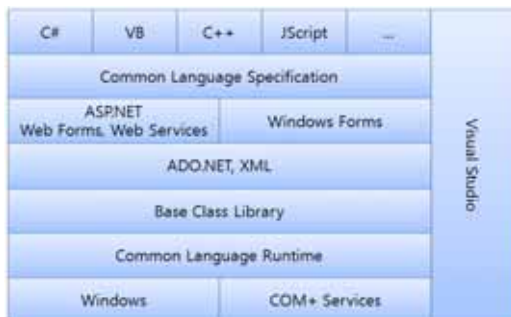
내부 구현을 모르더라도 운영 환경에 관계없이 언어 중립적 방법으로 사용될 수 있는 컴포넌트의 실현을 가능하게 하였다. 그러나 다른 컴포넌트 기술들과 마찬가지로 중요한 문제점이 뒤따른다. COM 컴포넌트를 개발하기 위해 언어 중립적인 인터페이스인 IDL(Interface Definition Language)로 인터페이스를 정의한 후 실제 구현에 있어서는 C++와 같은 개발 언어를 사용하여야만 한다. 결국 개발자는 작성하고자 하는 프로그램의 성격에 따라 또 다른 언어를 학습해야만 하는 경우가 종종 발생한다. 또한 컴포넌트의 사용자는 객체 모델의 표현을 위해 사용되는 형식 라이브러리에 의존하게 되는데, 만일 형식 라이브러리의 내용이 실제 구현과는 다르게 기술된 경우, 혹은 버전 관리의 문제로 인해 설치된 컴포넌트가 삭제될 경우 애플리케이션이 동작하지 않는 DLL Hell에 빠지게 된다[6].

닷넷(.NET)은 COM의 이러한 문제점들을 해결하고, 유사한 목적을 지원하면서도 서로 다른 형식으로 제공되던 내부적 기술들을 통합하고 단일화하며, 마이크로소프트와는 다른 기술을 선택한 외부 시스템이나 컴포넌트와도 효과적으로 상호 운용할 수 있는 플랫폼으로 개발되었다.

닷넷은 언어 중립적이고 안전한 실행 환경인 CLR(Common Language Runtime), 임의의 닷넷 기반 언어에서 사용할 수 있는 근래의 보편적 애플리케이션 도메인을 지원하는 표준 라이브러리 등으로 구성된 "닷넷 프레임워크(.NET Framework)"를 그 핵심으로 하고 있다[그림 1][7].

## 2. WPF

기술을 중요시하는 대부분의 소프트웨어 전문가들은 응용 프로그램과 사용자 간의 상호작용방식보다는 응용 프로그램의 작동 방식에 훨씬 더 많은 관심을 가집니다. 그러나 응용 프로그램을 구입하고자 하는 실제 사용자는 사용자 인터페이스를 보고 구매 결정을 하게 됩니다.



▶▶ 그림 1. 닷넷의 아키텍처

이처럼 응용 프로그램에 있어서 사용자 인터페이스는 해당 소프트웨어에 대한 전체적인 사용자 환경에서 매우 중요한 부분을 차지하며, 실제 사용자에게 있어서 이러한 환경은 응용

프로그램 '그 자체'를 의미합니다. 그러므로 더 나은 인터페이스를 통해 향상된 사용자 환경을 제공하게 되면 생산성을 높이고 우수 고객을 더 많이 확보할 수 있으며 웹 사이트에서의 매출을 늘리는 등 다양한 효과를 얻을 수 있습니다[8].

이전에는 문자 기반 인터페이스만으로도 데이터의 분석이 충분했지만 오늘날의 사용자들은 그래픽 인터페이스에 익숙해져 있으며, 사용자 인터페이스에 대한 요구사항은 계속해서 증가하고 있습니다.

웹의 발전은 소프트웨어와의 편리한 상호작용을 기대하는 사용자 계층을 형성하게 되었고, 사용자들이 응용 프로그램을 사용하는 시간이 늘어날수록 해당 응용 프로그램에 대한 인터페이스 요구사항은 점점 높아만 가고 있는 추세입니다. 이렇듯 점점 높아가는 인터페이스 요구사항에 부응하기 위해서는 사용자 인터페이스를 만드는 기술도 함께 발전해야 합니다.

Microsoft .NET Framework에 포함된 WPF(Windows Presentation Foundation)를 사용하면 이미지, 텍스트, 2차원 및 3차원 그래픽, 비디오 및 오디오 등을 포함하는 인터페이스를 만들 수 있습니다.

### 2.1 사용자 인터페이스를 제공하기 위한 통합 플랫폼

Windows 사용자 인터페이스를 만들기 위해서는 여러 가지 기술을 함께 사용해야 합니다[표 1].

개발자가 폼, 컨트롤 및 기타 Windows 그래픽 사용자 인터페이스의 일반적인 요소를 만들려면 대개 .NET Framework의 일부인 Windows Forms를 사용해야 합니다. 인터페이스에서 문서를 표시해야 할 경우 Windows Forms에서 부분적으로 지원되는 화면 표시 문서를 사용할 수 있지만 고정된 형식의 문서를 사용해야 할 경우에는 Adobe의 PDF 형식을 사용하거나 비디오 및 오디오 기능을 사용하려면 Windows Media Player를 통해 제공할 수 있습니다.

[표 1] 사용자 인터페이스를 만들기 위한 기술 비교

	Windows Forms	PDF	Windows Forms/GDI+	Windows Media Player	Direct 3D	WPF
Graphical interface, e.g., forms and controls	○					○
On-screen documents	○					○
Fixed-format documents		○				○
Images			○			○
Video and audio				○		○
Two-dimensional graphics			○			○
Three-dimensional graphics					○	○

이처럼 복잡한 과정은 실제 프로그램 작성에 적절하지 않다는 데 누구나 동의할 것입니다. 그러나 통합된 단일 솔루션인 WPF에서는 이 모든 것이 가능합니다. WPF라는 단일 기술을 통해 광범위한 기능을 제공하면 최신 사용자 인터페이스를 훨씬 쉽게 구현할 수 있다는 장점이 있습니다.

### 2.2 개발자와 디자이너의 공동 작업 환경

소프트웨어 전문가들이 많은 수의 응용 프로그램을 단독으로 개발하고 있지만 최상의 사용자 인터페이스를 구현하기 위해서는 전문 인터페이스 디자이너의 도움이 필요합니다.

그러나 개발자와 디자이너는 업무 진행 방식이 서로 다르기 때문에 두 분야의 전문가가 공동으로 작업하는 데는 여러 가지 문제가 있습니다. 디자이너는 만들기 쉽더라도 개발자는 구현하기 어렵거나 전혀 구현할 수 없는 이미지도 있습니다. 예를 들어 개발자는 기술적인 한계, 촉박한 일정, 기술 부족, 사소한 오해 또는 단순한 의견 차이로 인해 디자이너의 의도를 완벽하게 구현하지 못할 수 있습니다. 따라서 최상의 인터페이스를 만들기 위해서는 독립적인 두 분야의 전문가가 인터페이스의 품질을 떨어뜨리지 않으면서 공동으로 작업할 수 있는 환경이 필요합니다.

WPF는 이러한 공동 작업이 가능하도록 XAML (eXtensible Application Markup Language)을 사용합니다. 디자이너는 Microsoft Expression Interactive Designer와 같은 도구를 사용하여 사용자 인터페이스의 모양과 상호 작용 방식을 지정하여 XAML표현으로 생성하고, 개발자는 이렇게 생성된 XAML 설명을 Microsoft Visual Studio와 같은 도구로 가져옵니다[그림2]. 이렇게 하면 디자이너가 만든 정적 이미지를 토대로 인터페이스를 처음부터 다시 만들 필요없이 인터페이스 정의 자체를 그대로 가져올 수 있으며, 그런 다음 이벤트 처리기와 같이 인터페이스에 필요한 코드 및 응용 프로그램에 필요한 기타 기능에 해당하는 코드를 작성하게 됩니다.



▶▶ 그림 2. XAML의 동작

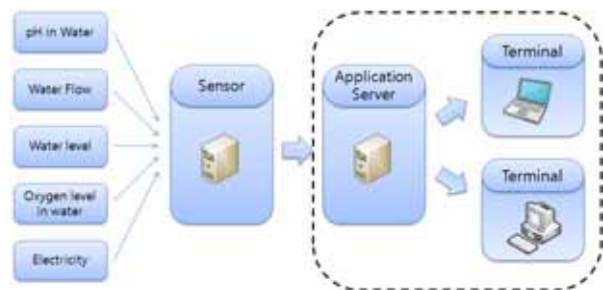
디자이너와 개발자가 이러한 방식으로 함께 작업하면 디자이너가 만든 이미지를 토대로 개발자가 인터페이스를 구현할 경우 발생할 수 있는 변환 오류를 줄일 수 있을 뿐만 아니라 효과적으로 의견을 교환하면서 동시에 작업을 진행할 수 있습니다.

### III. 실시간 환경 센서 데이터의 3차원 시각화 시스템 설계

환경 센서 모니터링 시스템은 각종 센서 및 측정 장비로부터 신호를 통해 환경정보를 전달 받고, 모니터링 센터는 각종 센서와 측정 장비로부터 전해 받은 자료를 분석하고 해석하여 기록하며, 단말기를 통해 사용자에게 해당 정보를 보여 주거나, 무선을 통해서 특정정보를 전달하기도 하며, 상황에 따라서는 경고 메시지를 보내게 된다.

#### 1. 환경 센서 데이터의 관리 시스템 구성도

그림 3은 각종 센서 및 측정 장비로부터 입력받은 환경 센서 데이터의 관리 시스템을 보여준다.



▶▶ 그림 3. 환경 센서 데이터의 관리 시스템 구성도

#### 2. 환경 센서 데이터의 시각화 시스템 구조도

그림 4는 각종 센서로부터 받은 다양한 데이터에 대해 시각화할 환경 센서 데이터의 시각화 시스템의 구조도를 보여준다.



▶▶ 그림 4. 환경 센서 데이터의 시각화 시스템 구조도

센서 측정기는 양식장의 pH, 유속, 수위, 용존 산소량 등의

측정값을 입력받는다. 시각화 시스템은 .NET Framework을 기반으로 하고 있으며, 첫 번째 단계로 사용자가 필요로 하는 데이터를 얻을 수 있도록 미정제된 숫자 데이터를 걸러낸다. 두 번째 단계는 선택된 데이터에 대해 의미 있는 데이터로 만들기 위해서 변형을 가하게 된다. 세 번째 단계는 일부 데이터에 대해서 3차원 데이터형태의 결과를 얻게 된다.

시각화 모듈 및 분석 모듈에서는 사용자의 요청에 의해 필요로 하는 자료를 분석할 수 있도록 이미지, 2D, 3D 그래프, 사운드, 비디오, 애니메이션 등의 형태를 통해 보여주게 된다.

### III. 실시간 환경 센서 데이터의 3차원 시각화 시스템 구현

#### 1. 시각화 알고리즘

사용자가 선택한 항목에 대해 상하좌우 회전 및 확대, 축소가 가능한 3차원 그래프를 통해 데이터를 분석할 수 있으며, 한계치를 넘는 센서 데이터 발생 시 경고 메시지를 화면에 띄운다.

표 2는 3차원 차트에 대한 회전 및 확대 기능 알고리즘이다.

[표 2] 3차원 차트의 회전 및 확대 축소 알고리즘

```
void DetailRotater3DTransition_RotateCompleted(object sender, EventArgs e)
{
    double angleTo = (double)DetailRotater3DTransition.GetValue(
        (IdentityMine.Sensor.Controls.Rotater3DTransition.
        AngleRotateToProperty);
    double angleFrom = (double)DetailRotater3DTransition.GetValue(
        (IdentityMine.Sensor.Controls.Rotater3DTransition.
        AngleRotateFromProperty);
    :
    detailMain.SetValue(Grid.ColumnProperty, (int)2);
    detailMain.SetValue(Grid.ColumnSpanProperty, (int)1);
    gridRelayout = true;
    MasterRotater3DTransition.SetValue(IdentityMine.Sensor.Controls.
    Rotater3DTransition.AngleRotateFromProperty, (double)30);
    MasterRotater3DTransition.SetValue(IdentityMine.Sensor.Controls.
    Rotater3DTransition.AngleRotateToProperty, (double)0);
    MasterRotater3DTransition.Rotate();
    double scaleTo = (double)MasterRotater3DTransition.GetValue(
        (IdentityMine.Sensor.Controls.Rotater3DTransition.
        ScaleXProperty);
    MasterRotater3DTransition.AnimateScaleXTo(scaleTo);
    double translateTo = (double)MasterRotater3DTransition.GetValue(
        (IdentityMine.Sensor.Controls.Rotater3DTransition.
        TranslateXProperty);
    MasterRotater3DTransition.AnimateTranslateXTo(translateTo);
    double scaleYTo = (double)MasterRotater3DTransition.GetValue(
        (IdentityMine.Sensor.Controls.Rotater3DTransition.
        ScaleYProperty);
    MasterRotater3DTransition.AnimateScaleYTo(scaleYTo);
    :
}
```

## 2. 모니터링 통합 화면

각종 센서 및 측정 장비로부터 전달받은 실시간 환경 센서 모니터링 시스템은 그림 5와 같이 4개의 모듈 단위로 구성된다.

### 2.1 위치 선택 영역

양식장별로 측정 장소를 선택할 수 있는 영역이다.



▶▶ 그림 5. 환경 센서 데이터의 시각화 시스템 구현 화면

### 2.2 2차원 차트 선택 영역

선택한 측정 장소별로 전달받은 환경 센서 및 측정 장비 데이터를 2차원 차트를 이용하여 실시간으로 표시해 주고, 경고 발생 시 경고 메시지를 출력해 주는 영역이다.

### 2.3 3차원 차트 표시 영역

측정 장소로부터 전달받은 각종 센서 데이터 및 측정 장비 데이터에 대해 3차원으로 보여주는 영역으로, 3차원 차트에 대해 마우스를 이용하여 상하좌우 회전 및 확대, 축소 기능을 이용하여 원하는 데이터에 대해 상세하게 데이터를 분석할 수 있다.

### 2.4 상세 표시 영역

환경 센서 및 측정 장비로부터 전달받은 데이터를 비롯한 양식장의 상세 정보를 표시해 주는 영역이다.

## IV. 결 론

본 논문에서는 양식장의 다양한 환경 센서 및 측정 장비로부터 전달받은 자료를 활용한 양식장의 환경 센서 데이터 시각화 시스템을 설계 및 구현하였다.

이 시스템은 사용자가 원하는 데이터에 대해 선택적으로 모니터링 할 수 있다. 예를 들면 용존 산소량, 수온, 수위, 유속 등 관심 있는 데이터를 선택하여 볼 수 있다.

이 시스템은 환경 센서 데이터를 실시간으로 모니터링 할 수 있으며, 한계치를 벗어나는 측정 데이터 발생 시 경고 메시지를 통해 사용자에게 알려주며, 실시간으로 변화하는 3차원 그래프를 통해 시각적 만족도를 한층 더 끌어 올렸다.

앞으로 이 시스템을 양식장 관리를 위한 모니터링 시스템에 적용할 계획이다.

#### 감사의 글

본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음(ITA-2007-C1090-0701-0001)

#### ■ 참고 문헌 ■

- [1] 성경환, 이대영, 김형구, “2D와 3D Graphic 기반으로 구성된 GUI의 효율성의 차이”, 한국콘텐츠학회논문지, 제7권, 제7호, pp87-88, 2007.
- [2] 이지수, 임창영, 권은숙, “멀티미디어 인터페이스 디자인의 평가 요소 분석”, 디자인학연구, Vol.11, No.1, pp35, 1997.
- [3] 진부향, “Web3D 기술 기반의 GUI 디자인에 관한 연구”, 중앙대학교 예술대학원, 2003.
- [4] O.I. Lekang, Odd-Ivar, and B.F. Eriksen, IT in Fish Farming, CIGR Handbook of Agricultural Engineering Volume VI Information Technology, pp. 325-328.
- [5] Don Box, Essential Com, Addison-Wesley International, 1997.
- [6] <http://msdn2.microsoft.com/en-us/library/ms811694.aspx>
- [7] 김명호 “마이크로소프트 컴포넌트 기술의 발전과 동향”, 한국정보과학회지, 제24권, 제11호, pp.5-6, 2006.
- [8] <http://msdn2.microsoft.com/en-us/library/Aa663364.aspx>