

Not All Design Methodologies are Created Equal

3 차원 모델링을 이용한 다양한 3D 설계 방식이 빠른 속도로 퍼져 나가고 있다. 따라서 어떤 3 차원 CAD 를 선택할 것인지를 결정한다는 것은 무척 어려운 과정이며, 특히 시장에 널리 사용되고 있는 선도적인 위치에 있는 CAD 업체들의 마케팅관련 메시지를 접한 상황에서는 더욱 혼란스러울 수밖에 없는 실정이다. 많은 회사들이 수년동안 이러한 혼란에서 벗어나 자신의 회사에 맞는 CAD 를 선택하기 위하여 나름대로의 의사 결정 과정을 수행하고 있다. 그러한 의사 결정 과정을 보면 어떤 회사들은 다양한 CAD 시스템들을 다각도로 면밀히 검토하여 결정하는 형태를 보이고 있다. 반면, 또 다른 회사들은 단순히 많은 사람들이 사용하는 CAD 시스템을 선택하는 형태를 보이고 있다. 이는 단순히 현재 가장 널리 사용되는 CAD 시스템을 선택한 그 많은 사람들이 모두 잘못된 결정을 하지는 않았을 것이라는 생각 때문일 것이다. 3 차원 CAD 시스템을 선택하는 모든 회사들은 이러한 양 극단의 중간 정도에 서 있다고 볼 수 있다. 어떠한 방법을 사용하던, 차기 설계 시스템을 선택할 때에는 현재 사용되고있는 다양한 CAD 시스템들은 나름대로의 다양한 설계 방법론을 채택하고 있다는 것을 먼저 이해하여야 한다. 또한 모든 설계 방법론들은 모두 동일하지 않다는 것을 반드시 고려하여야 한다. 각각의 설계 방법론들은 나름대로의 장점과 단점이 있기 때문에 생산하는 제품의 특성과 또한 설계 인력들의 문화에 적합한 것을 선택하는 것이 무엇보다도 중요하다. 잘못된 설계 방법론은 회사의 전체적인 설계 생산성에 지대한 영향을 끼치게 되며, 이는 곧 최종 제품의 경쟁력에도 그 영향을 미치게 된다.

다행히 솔리드 모델링에 대한 방법론은 솔리드 모델을 생성하는 기본적인 방식에 의하여 대표적으로 두가지의 방식 즉, parametric 과 non-parametric 으로 분리 되어왔다. 기본적으로 어떤 업체의 어떤 CAD 를 사용할 것인가를 결정하기 전에, 반드시 어떤 설계 방법론이 적합한지를 먼저 결정하여야 한다. 그런 다음에 결정된 설계 방법론을 채택하고 있는 CAD 시스템들 중에서 가장 적합한 것을 선택한다면 나름대로 CAD 시스템의 결정 과정에서 발생할 수 있는 오류를 최소화 할 수 있을 것이다. 이를 위해 두가지의 대표적인 설계 방법론에 대하여 간략히 살펴보기로 한다.

Parametric modeling

개 요:

Parametric Modeling 은 지오메트리를 정의하는 데 있어 일련의 지오메트리 상호 관련 파라미터 (dependent parameter)/구속(constraint)을 통하여 해당 지오메트리를 정의하는 방법으로서, 하나의 shape 를 수정하면 상호 관련(또는 구속)된 다른 shape 의 사이즈나 위치가 변하게 된다. 예를 들면, 어떤 face 의 중심에 있도록 만들어진 hole 은 일련의 수식을 통하여 생성된다. 따라서, 그 face 가 어떻게 바뀌던 그 hole 은 항상 자신을 그 face 의 중심에 위치시키

게 된다.

배 경:

PTC가 1980 년대에 동사의 CAD 시스템인 Pro/Engineer를 통하여 이러한 parametric modeling 방법론을 처음으로 소개하였다. 그 후, Microsoft의 Windows 환경이 도래하면서, 윈도우 환경에 맞도록 개발된 제품들이 SolidWorks, SolidEdge등이 있으며, 그 후 Autodesk에서도 같은 개념의 Inventor를 발표하였다. Parametric modeling은 현재 가장 널리 사용되는 설계 방법론이다.

Parametric design 방법론:

Parametric design 시스템은 사용자로 하여금 아주 정형화된 모델링 과정의 틀 안에서만 작업 하도록 되어있다. 사용자는 먼저 완전히 구속 조건들로 이루어진 2D 스케치를 먼저 그려야만 한다. 그런 다음에야 비로소 이 2D 스케치는 extrude 나 스핀등과 같은 여러 가지 모델링 조작을 통하여 3 차원 형상으로 변환된다. 이렇게 생성된 모델의 첫번째 feature 를 base feature 라고 한다. 그 다음, 사용자는 이러한 과정을 반복하여 base feature 에 추가적인 feature 들을 더해 나간다. 이때, 추가되는 각각의 새로운 feature 들은 기존에 생성된 feature 들 중의 하나에 위치나 사이즈가 구속되어야 하는 데, 이를 parent - child 구속 관계라고 한다. 이러한 parent - child 개념은 단순히 part 모델링에만 해당되는 것이 아니라, 모든 parametric 시스템이 동작하는 기반이기 때문에 소프트웨어의 모든 영역 즉, 어셈블리 생성, 컨스트럭션 및 데이텀 플레인 위치, 디멘전등 모든 동작에 영향을 미친다. 따라서, 어떤 모델에서 parent feature 가 수정되면 이에 영향을 받는 모든 child feature 도 변경되며, 이는 종종 원치 않는 결과를 만들어낼 수 도 있다.

Parametric modeling 의 장점:

Ability to capture design intent - parametric system은 하나의 feature가 다른 feature에 종속되어야 하기 때문에 디자인 의도를 확립하는데 매우 적합하다. 여기서 디자인 의도(design intent)란 어떤 형상을 만들고 끼워 맞추어 모델링을 하면서 기능적인 요구사항을 확인할 수 있는 것을 말한다. 이러한 것은 보통 여러 가지의 구속 조건들을 이용하여 수행할 수 있다.

3D capture of existing designs - parametric system은 기존의 이미 만들어진 디자인을 3 차원 모델로 만들어내는데 탁월한 성능을 보인다. 즉, 이미 파라미터등에 대한 내용이 확인된 그리고 변경이 없는 그러한 이미 확정된 디자인을 3 차원 형상화하는 데 탁월한 성능을 보인다.

Support for family of parts design - parametric system은 동일한 제품(parts) 군을 설계하는 데 매우 적합하다. 즉, 기본 제품에서 사이즈, 위치 등을 변경시켜 유사 제품을 쉽게 생성할 수 있다. 그러나 parametric system은 파라미터를 입력하여 새로운 추가적인 지오메트리를 생성

하는 경우에는 그리 성능을 발휘하지 못한다.

Parametric modeling 의 단점:

Rigid requirement of parent child relationships – parametric system은 모든 모델링 과정에서 상호 구속을 설정하여야 하기 때문에 사용자로 하여금 디자인 의도에 실제 필요한 구속보다도 더 많은 구속을 설정하도록 한다. 이는 곧 향후의 설계 진행 과정에서 여러 가지 문제를 일으킬 위험을 내포하게된다. Parametric system은 전술한 것처럼 어느 정도까지의 design intent를 구체화 시키는 데 효율적이지만(예를 들어 어떤 특정 edge로부터 항상 일정한 거리를 유지하며 정지하고 있는 실린더와 같이 그리 복잡하지 않은 경우), 만일 여러 개 예를 들어 수백개의 상호 구속을 더한다면 각각의 구속이 서로 쌓여 최종적으로 어떤 영향을 미치게 될지 예상하기 어렵게 된다 이러한 parent – child 구속 개념의 부정적인 영향은 향후에 예상치 못한 모델 수정을 야기할 수 있고, 더 나아가서는 모델 내부의 구속간의 상호 충돌로 인하여 수정에 따른 모델 재생성(regeneration)이 되지 않는 경우도 있다. 어느 쪽이던 간에 문제를 찾아 해결하려면 많은 시간이 소요되고 이로 인하여 예상했던 설계 생산성이 크게 저하된다.

Upfront knowledge of “Design Intent” requirement – parent - child 구속은 모든 parametric system의 기본 요건으로서, 일반적으로 design intent라고 알려진, 즉 모델이 어떤 기능을 할 것인지에 대해서 반드시 정확하게 숙지하여야 한다. 이러한 사전 지식(design intent)이 없다면, parametric system 사용자는, 창조적인 설계 프로세스를 제약하거나 어떤 경우에는 하나의 수정 사항 때문에 모델 전체를 재설계 해야 하는 최악의 상황까지도 초래하는 경우가 많이 있다. Hole을 예로 들어보자. 이 hole은 구속 조건을 통하여 특정 edge로부터 일정한 거리만큼 떨어져 있다고 가정한다. 이때 만일 기준이 되는 edge가 움직인다면 어떤 일이 벌어질까? 분명히 그 hole 또한 구속 조건을 유지하기 위하여 edge가 움직인 만큼 이동 할 것이다. 그러나 그 것이 사용자가 원하는 것이 아니라면? 그 hole의 원래 위치가 다른 이유로 인하여 매우 중요하다면?? 많은 경우, 설계자들은 창조적인 아이디어를 기반으로 새로운 모델들을 설계한다. 이러한 창조적인 설계 과정에서 앞으로 어떤 변경이나 수정이 필요할 지를 미리 모두 예상하여 안다는 것이 가능하겠는가??? 한마디로 이는 불가능하다. 여러 가지의 변경이나 수정 등을 통하여 새로운 아이디어를 실험하고 이를 통하여 새로운 모델을 창조하는 과정은 기본적으로 모든 수정 가능성이 열려있는 상황이기 때문이다. 설사, 비록 어떻게 수정될 것인지를 알았다 하더라도 복잡하게 얽혀진 parent – child 구속이 원하는 대로 동작하도록 모델을 구성한다는 것 자체가 그리 용이하지 않다. 모델이 복잡해 질수록 사용자의 창의적이고 혁신적인 아이디어가 구현될 가능성은 더욱 적어진다.

Requirement of replaying full history to view changes – parametric system은 각각의 feature들 사이에

순차적인 관계(sequential 즉, parent - child)를 만들어내기 때문에, 사용자가 어느 한 부분의 수정 사항을 적용한 결과를 보기 위해 경우에도 전체 history tree가 재구성되어야 한다. 작거나 간단한 모델의 경우에는 그리 큰 문제가 되지 않지만, 모델이 복잡해 질수록 히스토리 트리 재구성에 많은 시간이 소요되며 이는 곧 생산성과 직결되는 문제이다. 또한 이러한 문제는 어떤 수정 사항을 적용하여 히스토리 트리가 재구성되는 과정에서 히스토리 내의 어느 곳에서 문제가 발생하여 fail된다면 더욱 심각한 상황에 봉착하게 된다. 이 fail된 부분을 찾아 이를 해결하면 이로 인하여 또 다른 곳에서 예기치 못한 문제를 일으키게 되고 이러한 상황이 반복되어 문제의 해결에 더욱 많은 시간을 빼앗기게 되어 결국은 디자인 보다는 수정에 따른 문제 해결에 더 많은 시간이 소요되게 된다.

Increased model size decreases ability to innovate – 앞에서 언급한 parametric system의 단점들을 보면, 이러한 문제점들이 결국은 모델이나 어셈블리의 규모가 커질수록 더욱 증가하고 심각해진다는 것을 알 수 있다. 즉, 모델이 복잡해질수록 설계자의 창의성은 더욱 제약을 받게 된다. 아울러 기존에 만들어진 parent - child 구조의 예기치 않은 상호 간섭에 따른 문제 해결에 더욱 많은 시간을 뺏길수록 설계 생산성은 더욱 떨어지게 된다. 어떤 feature를 삭제할 경우에도 많은 제약이 따르는데 이는 parent - child 구조 개념을 통하여 다른 feature들이 삭제하려는 feature를 레퍼런스로서 참조하고 있기 때문이다. 또한 어떤 한 part의 위치를 바꿀 경우에도, 같은 이유로 인하여 다른 feature/ parts/assembly가 이동하려는 part를 레퍼런스로서 참조하고 있기 때문에 이 경우에도 많은 제약이 따른다.

Legacy model knowledge within company – parametric system 기반에서 parametric 모델을 생성하는 경우, 모델을 생성하는데 부여되는 parametric 구조에 대하여 완벽하게 이해하고 있어야만 향후에 설계 수정이 필요할 경우 이에 효율적으로 대처할 수 있다. 만일 이에 대한 충분한 지식이 없다면 설계를 진행해 가는 도중에 사용자는 분명히 위에 열거한 단점들의 희생양이 되고 말 것이다. 특히 팀을 이루어 어떤 프로젝트를 진행하는 경우에는 설계 초기부터 이러한 문제에 봉착하게 될 가능성이 더욱 농후하다. 예를 들어 엔지니어 B가 어셈블리 단계에서 엔지니어 A가 설계한 모델을 사용할 경우 과연 엔지니어 B는 엔지니어 A가 어떤 디자인 의도(design intent)로 그 모델을 설계했는지를 알 수가 있겠는가? 설계 과정에서 엔지니어 B가 엔지니어 A가 설계한 모델을 수정할 필요가 발생한다면? 더구나 엔지니어 A가 더 이상 그 프로젝트에 참여하지 않고 있거나 회사를 떠난 상황이라면??? 엔지니어 A와 B가 설계한 모델들을 이용하여 최종 어셈블리된 모델에 대하여 생산 기술부의 엔지니어 C가 생산 라인에 맞도록 설계 수정을 해야 한다면??? Parametric system은 여러 부서나 여러 사람들이 팀을 이루어 함께 설계 프로젝트를 진행하는 경우, 각각의 설계자들이 설계한 모델들의 모든 parent - child 구조에 대한 지식이나 이해가 모든 팀원들이나, 설계 다음 단계에 있는 생산이나 생산 기술 엔지니어들과 모두 완벽하게 공유된다는 것은 거의 불가능하다. 이러한 문제를 줄이려면 아주 엄격한 설계 지침이 필요한데, 이는 역으로 설계자들에게

게 시간적인 짐을 지우는 것과 동시에 설계에 있어서설계자의 창조성을 제한하는 요소로 작용하게 된다.

In context editing – 전통적인 parametric CAD 시스템들은 소위 feature tree history라는 것을 생성하는데, 이는 모델을 생성하기 위하여 각 feature들이 어떤 순서로 서로 연계되고 있는지를 나타내는 것이다. 이 파라메트릭 트리내의 어떤 feature를 수정하고자 한다면, 히스토리 내의 그 포인트 이후에 만들어진 모든 feature들은 숨겨지고, 단지 히스토리 내의 수정될 feature가 만들어진 포인트 까지의 지오메트리만 보여진다. 이러한 방식은 대단히 비 생산적인 상황을 초래하게 되는데, 그 이유는 사용자가 종종, 수정할 feature의 사이즈나 위치를 그 feature이후에 생성된(지금은 보이지 않는) feature를 기준으로 상대적으로 지정할 경우가 있기 때문이다.

Non-parametric modeling

개 요:

이 글에서는 non-parametric 모델링은 parent – child 구속을 사용하는 parametric 모델링을 제외한 모든 모델링 방법론을 총칭한다. 여기에는 일련의 기본적인 요소(analytic primitive)들을 더하거나 빼어 원하는 형상을 구현하는 불린(Boolean) 모델러가 포함된다. 또한, 새로운 face feature 를 더하거나 빼거나, 또는 단일 part 의 body 에 속하는 기존의 face 들을 수정하여 원하는 모델을 생성하는 direct b-rep 모델러도 여기에 포함된다.

배 경:

불린 모델러와 direct b-rep 모델러는 80 년대 후반에 나온 parametric 모델러 보다 훨씬 이전부터 존재 하였다. 현재까지 사용되고 있는 불린 모델러의 대표적인 예는 Autodesk 사의 MDT(Mechanical Desktop)과 CADKEY 사의 CADKEY 가 여기에 속한다. B-rep 모델러의 대표적인 예로는 CoCreate 사의 One Space Designer 가 있으며 현재에도 동사의 대표적인 제품으로서 판매 되고있다. CoCreate 사는 자사의 direct b-rep 모델링 기술을 다이내믹 모델링(Dynamic modeling)이라고 지칭하는 데, 그 이유는 어떤 모델의 수정에 있어 그 모델이 어떻게 생성되었는지에 관계없이 모델의 임의의 face 를 선택하여 수정할 수 있기 때문이다.

Non-parametric design 방법론:

non-parametric 디자인 시스템의 가장 큰 특징은 feature 히스토리를 유지할 필요가 없으며, 각 feature 들 사이에 parent – child 구속이 필요치 않다는 것이다. 예를 들어, 불린 모델러에서 블록을 관통하는 원통형 hole 을 위치시켰다면, 이 블록을 움직일 수 있는 단 한가지 방법은 먼저 원래의 hole 을 채우고 새롭게 하나를 만드는 것이다. B-rep 모델러의 경우에는, 먼저 원통형 hole 을 블록에 만들어 놓은 다음에는, 그 다음에 수정할 때 더 이상 hole 로 인식되지 않고 단일 b-rep 모델에 속하는 face 로 인식된다. 그러면, 이 hole 을 독립된 face 로서 자유롭게 움직일 수 있고, 만일 그 hole 이 part 의 다른 feature 와 교차(intersect)하게 되면, 각각의 part 에 남아있는 원통형 hole 의 각 부분은 또 다른 독립적인 b-rep face 가 된다. 따라서, 본래 원통에 속해있던 모든 face 들이 서로 연계되어 있다는 것을 알고있을 필요가 없다. non-parametric 모델러들은 히스토리 트리를 가지지 않기 때문에 parametric 모델러에서는 반드시 필요한 feature 재구성(re-ordering)과 같은 개념이 필요가 없다.

Non-parametric modeling 의 장점:

Guaranteed that what you edit is what you end up with – 늘 신경을 써야 하는 parent – child 구속이 없기 때문에 수정은 수정되는 face 에만 영향을 미친다. 따라서 말 그대로 WYSIWYG (What You See Is What You Get;보이는 그대로 만들어진다.) 수정이 가능한 패러다임이다.

Ability to edit geometry imported from other systems – 이기종 CAD 시스템간의 지오메트리 호환은 대부분 IGES 나 STEP 등과 같은 neutral format 을 통하여 이루어지기 때문에, 모든 feature 히스토리를 잃게 되며, 결국 단일(한 덩어리)의 b-rep shape 가 된다. 이러한 b-rep 은 direct b-rep 모델러에서 사용하는 기본적인 수정 방법이기 때문에, face 를 이용한 수정 방식을 통하여 아주 효율적으로 이러한 지오메트리를 다룰 수 있다. 반면, parametric 모델러는 항상 feature 인식(recognition)에 의존하여야 하기 때문에 이기종 CAD 시스템간의 파일 호환이 상대적으로 어려울 뿐 아니라 이기종 CAD 로부터 임포트한 지오메트리를 수정하거나 편집할 수 없다.

Faster regeneration times – Non-parametric 모델러는 모델 수정시 결과를 보기 위하여 전체 feature 히스토리를 재구성할 필요가 없기 때문에 일반적으로 복잡한 모델의 수정시 보다 빠른 결과를 얻을 수 있다.

Non-parametric modeling 의 단점:

Unable to maintain individual feature identity – feature 의 ID(identity)를 유지하면 특히 여러 개의 feature 들이 서로 교차하는 경우에 보다 빠르게 수정을 할 수 있다. 예를 들어, 블록을 뚫고 들어가는 많은 장홀(slot)들과 교차하는 실린더를 생각해 보자. Parametric 시스템에서는 아주 간단하게 실린더를 다른 위치로 옮길 수 있다. 이 경우 예전 hole 에 속하는 모든 face 들은 모두 사라질 것이다. 그리고 새로운 위치에서 새로운 hole 로 교체될 것이다. 반면, direct b-rep 모델러의 경우에는 원래의 실린더에 속하는 모든 face 들을 선택한 후에 이를 지정된 곳으로 옮겨야 할 것이다. 이렇게 모든 해당하는 face 들을 선택한다는 것이, 특히 보이지 않는 안쪽 부분의 face 들이 많은 경우에는 단순히 실린더를 옮기는 것보다 더욱 시간이 많이 소요된다. 불린 모델러에서는 기존의 hole 들을 메우기 위하여 솔리드 실린더를 더하고, 새로운 실린더를 새로운 위치에 만들어야 할 것이다. 따라서, parametric 모델러에서는 단순한 feature 이동이 non-parametric 모델러에서는 많은 시간이 소요된다.

Unable to re-order feature history – 적용된 feature 들의 순서를 바꿀 수 있는 parametric system 의 기능은 보다 빠르게 최종 지오메트리 결과를 바꿀 수 있다. 예를 들어, 이전에 셸(shell)을 가한 body 에 추가된 보스(boss)를 생각해 보자. 보스는 셸 다음에 추가된 것이기 때문에 내부가 차있는 솔리드일 것이다. 이때 보스도 셸을 주위위해서는 parametric 모델러에서는 단순히 보스를 히스토리 트리 내에서 셸 밑으로 옮기기만 하면 된다. 그러나 non-parametric 모델러에서는 이와 같은 단순한 방법으로 이를 실행할 수가 없다. 보다 복잡한 대안을 생각하던지 아니면 여러 번의 undo 를 거친 후 보스를 추가하고 다시 셸을 주어야 할 것이다.

More difficult to capture design intent – 모델을 설계하는 과정에서 설계자들은 디자인 의도 (design intent)를 유지하고자 한다. Parametric 시스템에서는 자연스럽게 이루어지는 것이지만, 구속 조건을 추가하는 2 차적인 과정을 통하여 디자인 의도를 확립할 수는 있을 것이다. 그러나 feature 들이 교차하는 경우에는 본래 feature 의 face 들을 여러 개의 분리된 face 들로 분해되기 때문에 이들이 모두 함께 움직이도록 구속을 주기가 어렵다.

어떤 디자인 방법론이 과연 나에게 맞는 것일까?

앞에서 두가지 디자인 방법론의 장점과 단점들을 살펴보았다. 이제 어떤 방법론이 과연 여러분과 여러분의 회사에 보다 적합한 것인지를 결정해야 될 시간이다. 대부분의 다른 사람들처럼 여러분도 똑 같은 의문을 가지게 될 것이다. “왜 이 모든 것들을 다 가질 수는 없는 것일까?? 두가지의 장점들을 취한 것은 없는 것일까??” 바로 그러한 의문에 대한 해답을 줄 수 있는, 두 가지의 장점을 취합한 CAD 시스템이 바로 **IronCAD** 이다.

IronCAD 의 태동 배경:

IronCAD 는 1998 년 캘리포니아의 산타클라라에 있는 Visionary Design System 이라는 회사에 의하여 처음 발표되었다. IronCAD 는 VDS 사가 1997 년에 합병한 3D/EYE 사의 기술을 기반으로 개발된 CAD 시스템으로서, CAD 의 새로운 패러다임을 제공한 혁신적인 기술로 인정받아 빠른 성장을 지속하고 있다. VDS 사는 1999 년에 회사명을 Alventive 로 바꾸고, 웹 기반의 유연한 디자인 협업 솔루션 개발을 목표로 하고 있으며, 2001 년 3 월에는 CAD 부문의 지속적인 개발을 위하여 기존의 IronCAD 관련 임직원들이 분사하여 IronCAD LLC 를 설립하였다. Alventive 로부터 분사한 IronCAD LLC 의 목표는 3 차원 디자인 툴로서의 IronCAD 의 지속적인 성능 개선하고, 이를 통하여 고객에게 최고의 설계 유연성과 생산성을 제공하는 것을 그 목표로 하고있다.

IronCAD 의 설계 방법론:

1990 년 후반에 IronCAD 가 개발된 이후, parametric 시스템의 장점과 단점이 알려지기 시작했다. IronCAD 개발팀의 목표는 parametric 시스템의 장점을 더욱 확대하는 한편 단점은 축소시키는 것을 목표로 하고 있었다. 이 목표를 달성하기 위하여, 서로 다른 방법론에서도 동작할 수 있는 보다 유연한 아키텍처가 필요하게 되었다. 이러한 아키텍처는 사용자로 하여금 해당 작업을 수행하는데 효율적인 방법론을 선택적으로 사용할 수 있다는 것을 의미한다. 항상 parametric 방법론에서 사용하던 엄격한 제한(parent – child 구속)이 더 이상은 필요 없게 되었다. 새로운 디자인 방법론으로서의 IronCAD 의 궁극적인 목표는 설계자의 선택의 자유와 폭을 넓혀주는 것이다.

이러한 혁신적인 설계 방법론의 주요 요소들은 아래와 같다.:

3F. SeoGyo B/D.,1576-6 Seocho-Dong, Seocho-Ku, Seoul, Korea.

Tel : (02)598-1623 Fax : (02)3471-9968

- Parametric CAD 툴과는 달리, 앞으로 어떤 변경이나 수정이 발생할 것인지에 대한 예측이나 염려없이 자유롭게 part 를 모델링 할 수 있다.
- 기존의 parametric 방법론처럼 2D 스케치에서 디자인을 시작할 것인지, 또는 기존에 만들어진 feature, part 또는 assembly 를 기반으로 디자인을 시작할 것인지, 작업의 특성에 맞도록 자유롭게 선택할 수 있다.
- 2D 스케치에 구속을 가할 것인지의 여부를 자유롭게 선택할 수 있다.
- 동일한 파일 안에서 여러 개의 파트나 어셈블리들을 생성할 것인지 아니면 별도의 파일에서 생성할 것인지를 자유롭게 선택할 수 있다.
- Feature 히스토리 트리를 생성할 것인지 아닌지를 자유롭게 선택할 수 있다.
- Part 수정시 feature 를 이용하여 수정할 것인지 아니면, direct face operation 을 통하여 수정할 것인지 자유롭게 선택할 수 있다.
- Feature/parts/assembly 등을 위치시킬 때, dynamic soft 로 할 것인지, 영구적인 구속 또는 구속을 주지 않을 것인지를 자유롭게 선택할 수 있다.
- 어떤 부분이 문제가 되었을 경우, 해당 part 의 feature 히스토리의 일정 부분만을 선택적으로 히스토리를 없앨 것인지를 자유롭게 선택할 수 있다.
- 히스토리 기반의 모델을 생성할 것인지 아니면 다른 형식의 모델을 생성할 것인지를 자유롭게 선택할 수 있다.
- 어떤 part 내의 어떤 feature 를 수정할 때, 비록 다른 feature 들이 히스토리 트리 상에서 해당 feature 보다 나중에 생성되었다 하더라도 자유롭게 수정할 수 있다.
- 수정에 따른 모델의 regeneration 시 발생 가능한 문제에 대하여 우려할 필요없이 자유롭게 구속 조건을 추가하거나 삭제할 수 있다.
- 단일의 어셈블리 내에 b-rep 솔리드 뿐 아니라 facet part (STL, VRML, 3DS)도 포함할 수 있다.
- 팀을 이루어 프로젝트 진행시 다른 팀원에 의하여 만들어진 파트나 어셈블리에 대해서도, 예상치 못한 수정 사항 발생을 염려할 필요없이 자유롭게 수정할 수 있다.
- 지속적으로 2D 스케치 플레인을 수정할 필요없이 dynamic handle 을 사용하여 feature 를 자유롭게 수정할 수 있다.
- 구속 값을 입력하거나 또는 구속 자체를 사용하지 않고도 feature/part/assembly 의 위치나 사이즈를 정확하고 쉽게 제어할 수 있다.
- 특정 프로젝트에 대하여 어떤 커널(ACIS 또는 Parasolid)을 사용할 것인지 자유롭게 선택할 수 있다.
- Parametric system 에서 모델 수정시 겪어야되는 많은 제약없이 모델들을 자유롭게 생성하고 수정할 수 있다.

Selecting the best tool for the job:

회사의 3 차원 설계 생산성을 극대화 시키는 가장 중요한 요소는, 회사의 제품 형태에 가장 잘 부합되는 설계 방법론을 기초로 하고있는 CAD 시스템을 선정하는 것이다. 그렇다고 해서, 소프트웨어의 디자인 방법론이 CAD 시스템 선정의 의사 결정 과정에 반영되는 단 하나의 조건만은 아니다. 여기서 논하는 것은 전체적인 의사 결정과정을 다루는 것은 아니고, 선택된 디자인 방법론에 의하여 조직에 어떤 영향을 미치는지에 대한 것이다.

여기서 기억해야 할 중요한 것은 모든 CAD 시스템들은 나름대로의 장점과 단점을 갖고있다는 것이다. 따라서, 어떤 CAD 의 어떤 장점이 곧 여러분이 늘 수행하는 작업의 형태에 가장 잘 부합하는 지를 찾아내는 것이 중요하다.

아래의 표(Figure 4-1)는 이러한 의사 결정 과정의 한 도구로서 사용될 수 있다. 사용자는 먼저 자신이 작업하는 내용 중에서 각 부문에 소요되는 시간을 %로 환산하여 기입한다. 그리고 디자인 방법론이 적혀있는 칸을 통하여 자신이 가장 많은 시간을 할애하는 작업 요소에 가장 강력한 장점을 가진 디자인 방법론을 찾아낼 수 있다. 이러한 방법은 전체적인 디자인 프로세스에서 가장 높은 생산성을 얻을 수 있는 3D 디자인 툴을 찾아내는 데 매우 유용하다.

Matching Design Activity to Design methodology

Design Activity	% time spent	Parametric	Non-Parametric	IronCAD
Parametric modeling		●	◎	◎
Family of part design		●	◎	◎
Moving known 2D designs to 3D		●	◎	◎
Modifying existing designs		◎	●	●
New assembly design		◎	◎	●
New part design		◎	◎	●
Un-anticipated change handling		◎	●	●
In-context part/assembly editing		◎	●	●

◎ 가능하지만 주요 장점은 아님

● 해당 디자인 방법론의 가장 주요한 장점

Figure 4-1

In Summary

3 차원 CAD 툴의 설계 방법론은 회사의 설계 생산성에 지대한 영향을 미칠 수 있다. 따라서, 작업의 성격에 맞는 적절한 설계 방법론을 선택하는 것이 매우 중요하며, 이를 통하여 설계 생산성을 크게 제고 시킬 수 있다. 각각의 설계 방법론은 태생적으로 강점과 약점을 갖고있기 때문에, 각 설계 방법론들이 각각의 작업 형태 중 어떤 부문에서 강점을 보이는지를 분석하여, 자신의 주요 작업 내용에서 강점을 보이는 방법론을 찾아내는 것이 새로운 CAD 시스템을 결정하는 과정에서 중요한 단계가 되고있다.

Parametric 설계 방법론을 채용하고 있는 시스템들은 모든 변경 요인들이 예측 가능하고 또 예상치 못한 수정이 발생할 가능성이 거의 없는 경우의 기존의 정형화된 디자인을 처리하는 데 매우 유용하다. 또한, parent- child 구속의 일부분으로서 파라미터 값을 입력하여 여러 part 들의 조합을 수정하므로써 동일한 패밀리 제품들을 설계하는 데 특히 유용하다. Parametric 의 속성상 모든 작업(명령)들이 상호 구속으로 이루어지기 때문에 디자인 의도 (design Intent: 어떤 형상을 만들고 끼워 맞추어 모델링을 하면서 기능적인 요구사항을 확인할 수 있는 것)를 구현하는 데 매우 탁월하다. 그러나 이러한 상호 구속을 통한 설계 방법론을 가진 parametric 시스템은 역으로 이러한 상호 구속 때문에 많은 수정이 발생하는 설계의 본 과정에서는 커다란 제약을 갖게 되어 오히려 이러한 부분에서는 단점으로 작용하게 된다. 특히, 다양한 창조적인 아이디어를 필요로 하는 초기의 개념 설계 단계와 설계 프로세스의 후단에 위치한 생산 수정 단계에서의 수정에는 커다란 제약을 갖게 된다. Parametric 모델러들은 예기치 못한 여러 가지 수정 사항들이 발생하면 이를 해결하기 위하여 악전 고투하여야 한다. 따라서, parametric 모델러를 이용하여 모델링하는 과정에서는 많은 것들을 사전에 고려하여야 하기 때문에, 이러한 parametric CAD 시스템을 학습하는데도 많은 어려움이 따르며, 이로 인해 일부 parametric cad 툴의 학습 곡선(learning curve)는 보통 수개월이 소요되는 경우가 보통이다. 또한 수정에 따른 모델 regeneration 시 오류없이 수정이 적용될 수 있도록 어느 정도의 숙련된 설계 작업 수준 까지 도달하는 데에는 많은 시행착오를 겪어야 하며 아울러 추가적으로 더 많은 기간이 소요된다. 즉, 설계 생산성의 향상을 위하여 사용하는 설계 도구의 사용법을 숙지하는 데에 너무 많은 시간과 노력이 요하게 된다.

IronCAD 는 새로운 설계 방법론으로서, **Freedom Architecture™** 를 채용하고 있으며, 이는 parametric 시스템의 장점을 수용하면서 그 단점은 최소화 시킬 수 있는 혁신적인 방법론이다. 즉, IronCAD 는 기존의 CAD 시스템들과 달리, 단 하나의 방법론이 아닌 다양한 방법론을 제공함으로써, 설계 과정에서 사용자가 작업의 특성에 맞는 설계 방법론을 선택할 수 있도록 구성되어 있다. 설계의 초기 단계에서 IronCAD 는 handle 기반의 기술을 이용하여

매우 개념적인 형태의 작업이 가능하다. 또한 디자인이 구체화 되어가면서, 사용자는 보다 더 세밀한 테크놀로지들을 사용하여 구체화된 디자인 작업으로 옮겨갈 수 있다. 이 단계에서 필요할 경우, 구속을 이용한 디자인 의도의 구현이나 다양한 형태의 자유로운 수정등을 통하여 모델을 설계할 수 있다. 이러한 다양한 그리고 상대적으로 자유로운 방법론을 통하여 IronCAD 는 혁신적인 방법론 및 획기적인 생산성을 제공하고 있다.

3 차원 CAD 시스템들은 시장에 도입된 지 꽤 오랜 시간이 지나 많은 진화를 거쳐왔기 때문에 오늘날 널리 사용되는 3 차원 CAD 시스템들은 주어진 작업의 95 ~ 98%를 작업해 낼 수 있다. 따라서, 새로운 3 차원 CAD 시스템을 선택할 경우에는 지엽적인 기능적인 면보다는 전체적인 면에서 어떤 시스템이 가장 높은 생산성을 제공해 줄 수 있는지에 그 초점을 맞추는 것이 현명할 것이다. 종종, 3 차원 CAD 시스템의 선정시 시장 점유율이 가장 큰 시스템을 선택하는 경우가 있는 데, 이러한 판단이 항상 옳은 것은 아니다.

어떤 설계 방법론이 회사에 가장 좋은 것인가를 선택하는 과정의 중요성이 종종 간과되는 경우가 있는 데, 대부분의 parametric CAD 시스템 공급사들의 경우, parametric 설계 방법론의 단점들을 얘기해 주는 경우는 그리 흔치않다. 따라서, 강점에 대한 부분만 주로 강조되기 때문에 단점에 대한 부분은 스스로 알아내는 수 밖에 없다. 이 글을 보면 parametric 시스템의 장점과 단점을 모두 알 수 있으며, 이를 통하여 최근 새로운 대안으로 급 부상하고 있는 IronCAD 의 혁신적인 패러다임을 확인할 수 있다.

기존의 많은 제품들과 비교하여, IronCAD 는 획기적인 생산성의 향상을 가져올 수 있다. 직접 IronCAD Trial 버전을 설치하여 확인하거나, IronCAD reseller 를 통하여 IronCAD 의 새로운 획기적인 어프로치를 직접 경험해 보면 기존 3 차원 CAD 시스템들과의 뚜렷한 차이를 쉽게 알 수 있을 것이다.

IronCAD 를 실행하고 모델링을 시작하는 순간, 왜 IronCAD 의 모토가 “**The Fastest and Easiest Way to 3D**”인지를 알게 될 것이다.