

“PROBLEM ARAŞTIRMASI”NDAN MİMARİ DEĞERLERE GEÇİŞTE BİNA PROGRAMLAMA

Pınar DİNÇ

Mimarlık Bölümü, Mühendislik-Mimarlık Fakültesi, Gazi Üniversitesi
06570 Maltepe Ankara, pinar@mmf.gazi.edu.tr

ÖZET

1970’li yıllarda ilk yazılı ürünlerini vermeye başlayan Bina Programlama/Mimari Programlama disiplini, günümüze kadar bir çok aşamadan geçmiştir. Bu aşamalarda değişimin temel izleği, salt pratik/uygulama içerikli dar bir veri alanını tanımlamaktan öteye geçip, “mimarlık” alanının tüm bileşenlerinin içerilebildiği ve bu bileşenlerin birbirleri ile etkileştirildiği geniş bir veri alanının tanımlanmasıdır. Bu çalışmanın konusunu oluşturan söz konusu geçiş, en belirgin ifadesini, Bina Programlama disiplininin dönüm noktalarında yer alan temel kitapların içeriklerinde bulmuştur. Çalışma içinde yer verilen her bir programlama kitabı, disipline, farklı bir tanım ve içerik kazandırmayı amaçlamaktadır. İşlevi gereği pratik alana doğrudan katılımı olan programlama disiplini, başlangıçta kendini *sistemik bir bilgi araştırması* olarak tanımlamış ve bilime yakın bir noktada konumlandırmıştır. En son ulaştığı noktada ise programlama, “mimarlık” alanının değerlere dayalı, dolayısıyla da aktörlere bağlı, yorum ve kabuller üzerinde temellenen ve her bir yeni yapıda yeniden tariflenmeyi gerektiren karmaşık ve çok bileşenli yapısına uyumlanmayı hedeflemektedir. Bu iki durum ve durumlar arası geçiş süreci, çalışmada, Bina Programlama disiplininin belli başlı kitaplarında yer alan disiplin tanımları ve programlama stratejileri üzerinden serimlenmektedir. Temel kitapların seçiminde, alana bir yenilik katma, bir dönüm noktası olma kriteri uygulanmıştır. Pratiği yönlendiren ve bir diğerinin söylemini tekrarlayan yapıtlar çalışma kapsamına alınmamıştır.

Anahtar Kelimeler: Bina programlama, mimari programlama, programlama yaklaşımları, programlama stratejileri, problem araştırması, mimari değerler

BUILDING PROGRAMMING: FROM “PROBLEM SEEKING” TO ARCHITECTURAL VALUES

ABSTRACT

Building/Architectural Programming discipline, which gave its first written works in

1970s, has been evolved gradually. The main theme of this change was; going beyond the limits of having a practice-based, narrow area and defining a broad spectrum that covers the whole constituents of the field of architecture, including the interactions between constituents. The change, which is the main theme of this study, shows itself in the contents of the main books of the field, most significantly. Each book, being reviewed in this study, aims to introduce a new definition and frame to the discipline. Being directly connected to the field of practice, architectural programming, was defined as a systematic search for information and placed by scientific research. In its ultimate position, architectural programming, aims to approach the complex and multi-layered structure of the field of architecture that depends on actors, therefore is based on interpretations and suppositions that has to be defined in each new case. In this study, these two positions, and the ones in-between, are reviewed in terms of the definitions of the discipline and the strategies introduced to the field. The choice of the main books have been made according to the “being a milestone/adding something new to the discipline” criterion. The books, which just guide the practice and repeat the existing discourses, are excluded in this study.

Keywords: Building programming, architectural programming, programming approaches, programming strategies, problem seeking, architectural values

GİRİŞ

Yaşam ve kullanım bütünlüğü için fiziksel bir çevre oluşturan, sayı ve içerik olarak zengin ve çoğu zaman da karşıt problemlerin bir arada çözümünü gerektiren mekansal organizasyonlar, belirleyicilik, dönüştürücülük ve simgesellik özelliklerine sahiptirler. Her mekansal organizasyon, kullanımı, insan davranışlarını ve ilişkileri (insan-insan/insan-mekan) belirlemekte, iklimi, kaynakları ve değerler sistemini dönüştürmekte ve ait olduğu kültürün değerlerini simgelemektedir [1]. Aslında bu özellikler, Vitruvius tarafından ortaya konmuş olan utilitas-firmitas-venustas üçlemesinin açılımlarıdır, bir anlamda bugünkü dilde ifadesidir. Dolayısıyla, mekansal organizasyonun, insanla doğal çevre arasında bir ara ortam olması ve kendi içinde insan eylemlerinin istenilen şekil ve düzeyde gerçekleşmesine izin verme durumu, hiçbir zaman değişmemiş ve değişmeyecek bir konumdur. Bu konum, mimarlık disiplinin karakteristik özelliğidir, onun sınırlarını ve etki alanını belirler.

Bu çerçeveden bakıldığında, her mekansal organizasyon, ister küçük bir oda (örn, bir konutta çocuk odası), isterse içinde farklı kullanıcı gruplarını barındıracak olan büyük bir organizasyonun mekanı/binası (örneğin, bir hastane, bir kampüs, bir fabrika) olsun, kullanıcı ile uyumu yakalamak, kullanıcılarının ihtiyaç, istek ve değerlerini barındırmak zorundadır. Çünkü, aslında her mekan bir çözümdür. Problemin tek ya da çok, basit ya da karmaşık, birbiri ile uyumlu ya da zıt parçalardan oluşması, mekanın problemler bütününe bir çözüm olması gerekliliği

durumunu değiştirmez. Sonuçta her mekan, yaşanılabilir (habitable) olmalıdır, insan eylemlerini, bireysel, grup ya da organizasyonel ölçeklerde desteklemelidir [2]. İnsan/grup ve çevre arasındaki uyum (fit), mekanın insana vereceği olumsuz etkilerin (stres, verimsizlik vb) azaltılması ve yaşanılabilirlik koşullarının değişen zamana göre yeniden tanımlanması [2] ile mekan(lar), insan(lar) için gerçek bir çözüm olabilmektedir. Dolayısıyla, yaşanılabilirlik, donmuş ve mutlak bir çözümü değil, yaşamla birlikte evrilen ve insan yaşamına doğrudan katılan, dinamik bir varoluşu ifade etmektedir. Doğaldır ki bu varoluşun bileşenleri, mekansal organizasyonlar ve kullanıcıların ihtiyaçlarıdır.

Mimarlık disiplini çerçevesinden bakıldığında, söz konusu dinamik uyumun sağlanmasında, programlama alanı, kilit noktadadır. Latince “*programma*” kelimesinden türemiş olan program, “*izlenilecek sıranın basılı/yazılı çerçevesi ya da özeti (brief) ; bir amaca doğru gerçekleştirilecek eylemlerin planı ya da sistemi ; yapılacak şeyleri ya da yer alacak etkinlikleri özellikle de kronolojik olarak listeleyen, formüle edilmiş bir plan*” [3] olarak tanımlanmaktadır. Kısaca program, gerçek kurgunun ne ve nasıl olacağı üzerine geliştirilmiş bir ön kurgudur. Bu nedenle, gerçekliğin bir ön bildirimi, bir modelidir. Modelin nasıl olması ve kendisinin nasıl kurgulanması gerektiği ise programlama disiplininin konusudur. Mimari programlama ya da tesis programlaması (facility programming) olarak da adlandırılan bina programlamada program, projenin hedeflerini, hedeflere ulaşmak için kullanılacak gerçekçi araçların neler olduğunu, projenin kapsamını, gidiş yönünü, bir sonraki aşama olan tasarımın içinde yapılacağı bağlamın niteliklerini ve sonuç ürünün kalite özelliklerini, işverenin ihtiyaç, istek ve değerlerini barındıran yasal bir belge olarak tanımlanmaktadır [4,5]. Programlama ise, tüm bu maddelerin yazılı ve yasal hale getirilmesi sırasında izlenen araştırma, dökümantasyon ve iletişim eylemlerinin tümünü kapsamaktadır.

Yukarıdaki tanımlamalarda, program ve programlama, bir tasarım öncesi süreç olarak ele alınmakta ve tasarıma yön verici bilgilerin toplu halde bir araya getirilmiş bir bütünü ifade etmektedir. Bunun yanı sıra, tasarımla eşzamanlı gerçekleşen programlama yaklaşımları (lineer olmayan/entegre/just-in-time yaklaşımı) [6] ya da “*programlama tasarımıdır*” [7] sloganıyla ifadesini bulan görüşler de mevcuttur. Aslında bu görüşler, programlama ve tasarımın birbirinden ayrılmaması/birbirini dışlamaması gereken süreçler olması tercihini dile getirmektedir. Zira programlama, gelişmiş bir yaşanılabilirlik için en gerekli süreç olarak ele alınmaktadır [2]. Tasarımla entegrasyon, programlama disiplininin ilk ortaya çıkışından günümüze kadar geçerli bir tercih olmuş, ancak entegrasyonun nasıl olması gerektiği farklı kuramcılar tarafından farklı şekillerde ele alınmıştır. Programlama yaklaşımlarının farklılaşmasını doğuran başlıca etmen, tasarımla entegrasyondur. Bir anlamda, programlama disiplini, mimari tasarımın bütünü içinde, kendi disipliner gelişimini sürdürür gibi görünmekle birlikte, aslında tasarımla olan bağlarını ve ilişkisini yeniden yeniden sorgulamaktadır. Bu çalışmanın konusu da söz konusu sorgulamada dönüm noktalarını belirleyen temel görüşlerin serimlenmesidir.

ÇALIŞMADA İÇERİK VE YÖNTEM

Mimarlık disiplininin çok katmanlı (multi-layered), değerler (values) üzerine kurulu doğası ve bu doğanın merkezinde, kendi özgül yapısıyla yer tutan tasarlama pratiği ile bütünleşme yolunda, bina programlamanın geçirdiği evrelerin açılımı, bu çalışmanın içeriğini oluşturmaktadır. Amaç, bina programlamanın bilimsellik temeline dayalı pragmatik yapısının, rasyonel stratejilerin ötesine geçip, *değerler* olarak adlandırılan ve tek defaya özgüllükleri ve irrasyonellikleri de barındıran doğal/organik ve esnek bir yapıya geçişin serimlenmesi, bu yolla mimarlık içinde yer alacak tüm alt disiplinler için bu tip bir organiklik ve esnekliğin gerekliliğinin vurgulanmasıdır. Bunun için, bina programlama disiplini içinde, izi sürülmeye çalışılan bağlamda önemli dönüm noktalarına karşılık gelen kuramsal ve derleme örneklerin yer aldığı yapıtlar, programlamaya getirdikleri tanım ve içerik yenilikleriyle ele alınmıştır. Her bir yapıtın, disiplinin organikleşmesi ve esnekleşmesi yolundaki duruşları (yanında/karşısında) saptanarak serimlenmiştir.

BİNA PROGRAMLAMA DİSİPLİNİNİN DÖNÜŞÜMÜNDE ADIMLAR

Yukarıda genel çerçevesi çizilen dönüşüm, başlıca üç adımda gerçekleşmiştir. İlk adımda, bir tasarım öncesi hizmeti olarak ele alınan, fakat tasarımdan ayrı tutulan programlama, kullanılması önerilen araştırma adımları ve teknikleri, bulguları sınıflama ve ifade etme biçimleri ile bilimselliğe yakın bir konumda ele alınmıştır. İkinci adım, ilk adımın pratiğe yansıtılması sonucu ortaya çıkan çözümleri, içerik olarak çeşitlenmeyi ve tasarımla bütünleşmeye geçişi ifade etmektedir. Son adımda ise, programlama disiplini, tamamen tasarımın ve mimarlık alanının çok bileşenli yapısına uyumlanmayı amaçlar hale gelmiştir. Bir başka anlatımla artık, tasarıma ya da mimarlığa, kendine ait ayrı bir çerçeveden bakan ve girdi sağlayan ayrık bir alan olmaktan çıkmış, sürecin doğal ve diğer alt süreçlerle uyumlu bir parçası haline gelmiştir. Aslında bu durum, bina programlamanın başlangıcından bu yana gündeminde yer alan bir özelliktir. Uygulamaların nicel olarak artması ve nitelik olarak çeşitlenmesiyle, örtük bir yan tema olmaktan çıkarak, disiplin için bir ana tema/izlek haline gelmiştir. Bilindiği gibi “*bina oluşum süreci*” (building delivery cycle) olarak da adlandırılan bütüncül süreç, planlama, programlama, tasarım, üretim, kullanım ve değerlendirme alt süreçlerini içermektedir. Her bir alt süreç, geri-beslemelerle, bütün bir sürecin gidişatını belirlemede, diğer alt süreçlerle eş bir etkililik alanına sahiptir [2].

Söz konusu adımları belirleyen eserler, disipline bakışları ve geliştirdikleri stratejiler özelinde aşağıda ele alınmıştır. Eserlerin seçiminde, bir önceki söylemden farklı bir alanı tanımlama ve disiplinin gelişim çizgisine yeni bir boyut katma kriterleri göz önünde tutulmuştur. Kuşkusuz, örnekler çoğaltılabilir ve inceleme derinleştirilebilir. Bu çalışmada, bir makale ölçeğinin korunması amaçlanmıştır.

1. Adım: Başlangıçlar

Bina programlamanın ilk kuramsal kitaplarını içermektedir. Alanı, uygulaması ve sürekliliği olan bir disiplin haline getirmek amacıyla ortaya konmuş başlıca iki yapıtı vardır. Pena’ya ait ilk yapıtın içeriği, programlamanın ne olduğu, sonuç ürünü (binayı) hangi başlıklar altında ele aldığı ve programlama prensipleri üzerine kurulmuştur. Programlamayı, bir *problem arama* davranışı olarak yorumlar. Fiziksel ve içerik olarak, bir başlangıç/tanıtım kitapçığı ölçeğindedir. Temel kavramların tanımları, programlama strateji ve tekniklerinin açıklamaları ve örnek programlar, asıl metinden daha uzun olan, “*ekler*” bölümünde yer almaktadır [8]. Sanoff’a ait ikinci yapıtın içeriği ise bilimsel olmaya ve tasarımla bilimsel bir bakış altında birleşmeye daha yakın bir konumdadır. Programlamayı, bir *problem çözme* davranışı olarak ele alır. Tasarımcılara, programlamanın içeriğini tanıtmayı amaçlayan bu eserde Sanoff, Pena gibi kendine ait bir model önermemekte, kullanılabilecek temel kavram, bilgi toplama, işleme ve dönüştürme tekniklerinin bir dökümünü yapmaktadır [9]. Eserlerin, çalışma bağlamındaki detaylı açıklamaları aşağıdadır.

1.1. Pena: Problem Seeking (1977)

Pena, kitabın üçüncü baskısına yazdığı önsözde, mimarlığın bir bilim olduğu kadar bir sanat olduğunu da belirtmekte ve her iki durumda da mimardan yaratıcı olmasının beklendiğini çizmektedir. Ona göre, mimar, bilimsel yaratıcılığında, sanatsal yaratıcılığında olduğundan daha disiplinli ve düzenli olmalıdır. “*Bilimsel bir metot, sistemli bir bilgi takibidir. Programlama, sistemli bir veri araştırması olduğunda, en iyi durumunu alır*” [8]. Bu çerçevede Pena, programlamanın ne olup ne olmadığına dair de bir liste geliştirmiştir. Pena’ya göre, programlama hüristik, analitik bir problem arama sürecidir, tasarım ise algoritmik, sentezci bir problem çözme sürecidir ; programlama tasarım değildir, fakat tasarıma giriştir (prelude); programlama, özü ve daha içerikli bir çözümü bulmaya yönelik, ihtiyaçların isteklerden ayrıştırıldığı, sınırlılıkların ve olanakların serimlendiği ve her bina türü için aynı olan, son derece gerekli bir süreçtir; programlama tüm bir problemin ne olduğunun bulunmasını içeren, şimdiki ve gelecek ihtiyaçlara karar veren ve salt istenilen mekanların listesi olmayan, yüksek düzeyde iletişim gerektiren bir ortamdır.

Bu tanımlamalardan anlaşılmalıdır ki, Pena için programlama, kesinlikle tasarımdan ayrı, farklı düşünme ve araştırma metotları gerektiren bir alandır. Tasarıma başlangıç olarak ele alınması, bir entegrasyonu işaret etmekle birlikte, bunun nasıl olacağına dair açık bir çözüm sunmamaktadır. Kitapta vurgulanan *şematik tasarım* kavramı, temel mimari kavramların, mekan ihtiyaç ve ilişkilerinin, temel sirkülasyon, ölçek, kitle, arazi kullanımı, genel görünüş, projenin kapsamı ve bütçeye uygunluğunun yazılı ve şematik olarak ifade edildiği, üç aşamalı (1. programlama, 2. şematik tasarım, 3. tasarım geliştirme) düşünülen tasarım sürecinin ikinci adımıdır, fakat yine programlamadan tamamen ayrı olduğu, kitabın

ekler bölümünde yer alan sözlükte, Pena tarafından vurgulanmaktadır. Şematik tasarımın içeriği irdelendiğinde bu vurgu, programlamayı, içeriği ne olursa olsun, isminde tasarım olan her başlıktan ayrı ve öncelikli tutma önyargısının bir sonucudur. Gerçekte, Pena'nın üç aşamalı tasarım süreci yaklaşımında ayrışma, programlama-şematik tasarım birlikteliğinin oluşturduğu süreç ile tasarım geliştirme süreci arasında olmalıdır. Yani, şematik tasarım, programlamanın doğal bir parçası/uzantısıdır.

Pena, ortaya koymuş olduğu bu ayırım sonucunda, programı daha çok yazılı bir belge olarak görmektedir. Bu belgenin hazırlanışı beş aşamalıdır: (1) hedeflerin belirlenmesi, (2) olguların toplanması ve analizi, (3) kavramların açıklanıp test edilmesi, (4) ihtiyaçlara karar verilmesi ve (5) problemin ifade edilmesi. Kitabın ek bölümünde yer alan örnekler incelendiğinde, problem ifadelerinin dört faktörlü (işlev, biçim, ekonomi ve zaman) yazılı ifadeler olduğu görülmektedir. Olguların toplanması ve analizi aşaması ise 24 adet program kavramı (örn; mahremiyet, hiyerarşi, karakter, yoğunluk, hizmet grupları vd) üzerinden gerçekleşmektedir. Pena, kavramları, müşterinin ne istediğini ifade eden bir dil oluşturması nedeniyle, programlamanın kalbi olarak görmektedir. Tanımını ise şöyle yapmaktadır: *“Program kavramları, müşterinin performans problemlerine fiziksel çözümler bulmadan, işlevsel çözümleri işaret eden soyut fikirlerdir... örneğin örtü bir program kavramıdır, karşılık geldiği tasarım kavramı ise çatıdır”* [8]. Kitapta, her kavram, bir grafik anlatımla desteklenmiştir.

Bu durumda Pena'nın, disipline ait bir ilk ürün veriyor olmanın ve bunu da bilimin sınıflandırıcı/kategorize edici bakışı altında gerçekleştiriyor olmasının da etkisiyle, mimarlık alanının ancak belli bir bölümünü kapsayan bir anlayış ortaya koyduğunu söylemek yanlış olmayacaktır. Bu bölümün, tasarıma hizmet etmek için var olan, ancak eylemlerinde onun farklı bir yaratıcılık alanının kapsamına giren sorumluluk alanına dahil olmamaya çalışan, sınırları özenle çizilmiş bir bölüm olduğu açıktır. Pena'nın çizdiği çerçeveye nelerin eklendiği ve kapsamın nasıl geliştirilip esnekleştirildiği, sonraki dönemlere ait kitapların içerik irdelemelerinde, daha belirginlik kazanacaktır.

1.2. Sanoff: Methods of Architectural Programming (1977)

Sanoff, kitabının ilk bölümünde, insan-çevre ilişkisinin önemine değinerek, kullanıcıyı da yoğun bir biçimde bu değişimin içine katmanın gerekliliğini vurgulamaktadır. Sanoff'a göre fiziksel çevre çok-disiplinli (multidisciplinary) ve çok-etkili (multieffect) bir ortamdır ve bu ortam ancak süreçlerde daha sistemli olunarak yaratılabilir. Bu nedenle programlamaya ihtiyaç vardır. Sanoff, bu düşünce biçimini, kökeni Descartes'e dayanan sistem teorisi ya da sistematik düşünme alanlarına dayandırır [9]. Buradan anlaşılacaktır ki, Sanoff da, Pena gibi, programlamaya bir sistematiklik, bir düzen açısından yaklaşmaktadır. Bu bağlamda,

program da, bilginin sınıflandığı ve bir sisteme bağlandığı (codification), tasarım sürecini sistematize eden, düzenleyici bir belgedir.

Aynı bakış açısının bir uzantısı olarak, tasarım da yedi aşamalı bir süreç olarak tanımlanmaktadır: (1) Bir problemin varlığını görmek ve onu çözmek için bir şeyler yapmanın gerekliliğinin farkında olma, (2) Problemin doğası hakkında karar verme, onu tasarımcının kullanabileceği bir problem ifadesine dönüştürme, (3) Problemin çözümüne yönelik tüm bilgilerin toplanması, çözüme yönelik değişkenlerin tanımlanması, (4) Tanımların, bir örnek üzerinde sınanması, verinin geçerliliğinin sınanması, (5) Bir çözümün seçilmesi ve değerlendirilmesi, (6) Çözümün aktörlere ve çeşitli katılımcılara sunumu, onlar tarafından yorumlanması ve (7) Kullanıcı memnuniyeti üzerine elde edilen bilginin gelecek tasarım problemlerinde kullanılması. Gerçekte bu süreç, hem programlama hem de tasarım için kullanılabilir teorik bir düşünce ve uygulama stratejisini barındırmaktadır. Dolayısıyla, salt tasarıma ait olduğu iddia edilemez. Bir başka anlatımla, Sanoff için tasarım ve programlama, aynı sistematik düşünce biçimine dayandırılabilen, benzer alanlardır. Zira, Sanoff programlama için de (1) mevcut bilgiye yenilerinin eklendiği bir genişletme (expansion), (2) problemin alt parçalarına ayrıldığı bir sınıflandırma ve (3) alternatifler arasından en iyisinin seçildiği bir değerlendirme sürecini önermektedir. Buradaki düşünce ve uygulama mantığı, tasarım için önerilenden farklı değildir.

Bir düşünce biçimindeki ortaklıktan öte, Sanoff, programlamayı tasarımla birbirine bağlamanın, başka türlü yollarını da aramıştır. Bunun için, kullanıcı verilerinin (user information) hangi yöntemlerle toplanabileceğine ilişkin bir döküm yaptıktan sonra, *tasarım verisini dönüştürmek* ile ilgilenir. Kitapta dökümünü yaptığı tüm tasarım metodlarının, temelde, amaçları için kullanıcıyı merkez aldığı belirir. Tüm modellerin tasarım sürecine daha taze bir bakış sağlamak için bir dizi bilgi toplama yöntemine ihtiyaç duyduğunu vurgular. Kuşkusuz, burada sözü edilen bilgi toplama süreçleri programlamanın bilgi toplama süreçlerinin tam karşılığıdır. Bir başka anlatımla, programlamayı tasarımın içinde, tasarıma veri sağlayan bir alt süreç olarak görür ancak bunu açık olarak ifade etmediği gibi aralarındaki bağı Pena'nın da ele aldığı gibi lineer bir bağ olarak yorumlar. Yani, bilgi toplama süreçleri biter, tasarlama başlar. Yer verdiği tasarım yaklaşımlarının (sistematik tasarım, C. Alexander'in pattern language yaklaşımı, eylem analizleri yaklaşımı, bina performans sistemi yaklaşımı ve morfolojik yaklaşım) seçiminde ve açılımında, her tasarım sürecinde kullanıcı verilerine duyulan ihtiyacın altı daha kalın çizilir. Kitabın son bölümünde dökümü yapılan programlama yaklaşımlarının dökümünde de, mevcut yapılardaki kullanım ve kullanıcı özellikleri üzerinden gerçekleştirilen çalışmalar ağırlıktadır. Dolayısıyla, programlama, mevcudun kullanım ve kullanıcı üzerinden değerlendirilmesine -çeşitli değerlendirme teknikleriyle- dayandırılmaktadır. Benzer değerlendirme tekniklerine, Pena da kitabının ekler bölümünde bir döküm olarak yer vermiştir. Sanoff'un bir katalog oluşturma çabası, programlamayı oluşturan her konuya eşit derecede önem vermesini gerektirmiştir.

Genel olarak, Sanoff’un programlamaya bakışı, iki temel saptamayla özetlenebilir: (1) programlamayı oluşturan düşünce biçimi ve uygulama süreçleri, tasarımı oluşturan düşünce biçimi ve uygulama adımlarıyla ortaktır ve (2) her tasarım yaklaşımında, adı geçmemekle birlikte, örtük olarak yer alan bir programlama çalışması mutlaka vardır. Bu varlığın nedeni, 70’li yılların tasarım teorilerini biçimlendiren, tasarımcının diğerlerinin adına karar verdiği geleneksel yaklaşımın tam tersini savunan ve tasarımcının mekansal yaratım gücünün önemli bir parçasını, kullanıcı verilerinin oluşturması gerekliliğine olan inançtır. Dolayısıyla, altta yatan temel paradigma, kararların müşteri ve kullanıcıyla birlikte alınmasını sağlayan, *katılımdır*.

2. Adım: Çözümler -ya da- Geçiş

1980 ve 90’lı yılların başlarında, bir önceki aşamada sistemlilik ve bilimsellik üzerine kurulu olan bina programlama anlayışı, katılım paradigmasının güçlenmesi -kullanıcıya olan ilginin yoğunlaşması- ve bilimsellik savlarının -uygulamalar sayıca artıp, içerik olarak çeşitlendikçe- başlangıçtaki belirleyici rolünü kaybetmesi sonucunda, çözülmeye başlamıştır. Bu dönemde, kılavuz kitaplar kadar, örneklerin yer aldığı derlemelerin gündeme geldiği gözlenir. En önemli kırılma, bir önceki dönemde, kullanıcı verisi olarak üzerinde durulan bilginin, bu dönemde ve sonrasında, *değerler* olarak adlandırılmasıdır ve bu isim değişikliğinin bir içerik değişikliğini de beraberinde getirmesidir. Bu çalışmada, bu geçiş dönemi, başlıca üç kitap üzerinden örneklenecektir. Bunlardan, derleyiciliğini Preiser’in yaptığı ilk iki kitap [10,11], farklı alanlardan örneklerin bir araya getirilmesini içermektedir. Bu derlemelere, özellikle Preiser ve Hershberger’in yazmış oldukları kuramsal bölümler, değerlerin programlama alanına katılımının ve programlamanın tasarımıyla daha sıkı entegre oluşunun somut ifadeleridir. Hershberger’in bu kitaplardaki yaklaşımı, bir sonraki aşamada, değerler üzerine kurulu ve kendine özgü programlama yaklaşımının temelini oluşturmuştur. Duerk’in yazmış olduğu kitap [4] ise, yine değerleri merkeze koyan ve o zamana kadarki literatürün kullanılabilir bir dökümünün yapıldığı, yazara ait ve uygulamaya dönük/pragmatik bir yaklaşıma dönüştürüldüğü, örnek programlara da yer veren kapsamlı bir eserdir. Söz konusu eserlerin, çalışmanın bağlamı için önemli olan noktaları aşağıda özetlenmiştir.

2.1. Preiser (ed): Programming the Built Environment (1985)

Kitap için yazdığı önsözde Preiser, programlama için kapsamı ve etkililiği geniş bir tanımlama yapmaktadır. Bu tanıma göre, programlama “*bir organizasyonun, grubun ya da bireyin misyonu ve hedefleri konusunda bilgi sağlayan ve bunları entegre olmuş eylem-insan-eşya ilişkilerine sistematik olarak tercüme eden, bu yolla da etkili, işlevsel bir bina ya da tesis elde eden*” bir süreçtir [12]. Dikkat edilirse tanımda, bilimsel ya da bir disipline ait alanı tanımlamaya yönelik yaklaşım geri çekilmiş, işlevselliği olan ve insandan hareket eden bir yaklaşım ön plana çıkmıştır. Programlama tanımında yer alan “*Bir program, bir organizasyonun mevcut*

durumundaki organizasyonel değişimleri ya da işlevsel farklılaşmalarını, yeni bir tasarım ya da bina ortaya koymadan da gösterebilir” [12] ifadesi, programlamanın, bir çevredeki yaşamı ve bu yaşamdan duyulan memnuniyeti yakından takip eden bir eylem alanı olduğunu işaret etmektedir. Yaşantıya ve kullanıcıya olan bu ilgi, disiplinin, Çevre-Davranış Çalışmaları ve Kullanım Sürecinde Değerlendirme yaklaşımıyla da yakından ilgili olmasını beraberinde getirmiştir. Bu yakınlıklar, programlamanın tasarım için olan önemini ve onunla olan iletişimini arttırmıştır.

Kıtapta ikinci bir giriş gibi yer alan Hersberger’in metni ise, *Mimari Programlama İçin Teorik Bir Altyapı* başlığını taşımaktadır ve bu alt yapının temel taşı değerlerdir. [13] Değerler devreye girdiğinde, “*programlama tasarımıdır*” ifadesi de gündeme gelmektedir. Çünkü, gerek programlamayı gerekse de tasarımı yöneten temel güç, aslında bir değerler bütünüdür. Bir anlamda, Sanoff’un programlama ve tasarımı bir düşünce biçiminde buluşturan bir önceki döneme ait yaklaşımına karşılık, Hersberger, düşünsel değil yaşamsal olan bir başka alt yapıyı/değerleri merkeze almaktadır. Hersberger için değerler, işverenin hedeflerini, kullanıcının ihtiyaçlarını ve mekansal gereksinimleri oluşturmadan önce ele geçirilmesi gereken temel bir bilgidir. Altta yatan değer sistemlerinin (value systems) çözümlenmemesi durumunda, hedef ve ihtiyaçlara yönelik tüm tanımlamalar da mesnetsiz kalmaktadır. Tüm bu söylemin altında yatan bir başka kavram *kaliteli mimarlık* (quality architecture) kavramıdır. Hersberger, yarara yönelik, işlevsel ve etkili olmanın ötesine geçen ve sanat alanına katılabilen bir eylemi, kaliteli mimarlık olarak tanımlamakta ve programlama disiplininin, mimarlığın şüursel boyutuna da bir kapı aralığı açmaktadır. Bu durumda programlama, değerlerin ve hedeflerin bir birine eklenildiği ve yüksek kaliteli bir mimarlığı işaret eden bir süreç olmaktadır. Bu süreçte, programlamacının, tasarımcıdan farklı bir kişilik olmayıp tasarımcının kendisi olması da önerilmektedir. Değerleri merkez alan bu yaklaşımın ismi de “Mimari Programlama”dır.

Bu noktada, kalıcı, kurumsal ve duruma yönelik başlıkları altında yer alan değerlerin, Hersberger tarafından kurulan teorik yapısının pratiğe yönelik bir karşılığının metin içinde yer almadığını ifade etmek gerekmektedir. Ancak, bu tip bir karşılık, aynı kitapta, “*gizli programın yeniden strüktüre edilmesi*” olarak isimlendirilen bir çalışmayla örneklenmektedir. [14] Bu çalışmada, *gizli program* kavramı, altta yatan değer sistemlerinin bir karşılığı olarak kullanılmıştır ve bir alış-veriş merkezinin, kullanıcıları tarafından bir süpermarket olarak değil, fakat bir pazar yeri olarak görülme istendiğine dair bir çalışmadır. Yani, çevre için altta yatan temel değer, mal alma ve satma değil, toplumsal olarak bir araya gelmedir. Bu tip bir değer alt yapısı olmadan pazar yerine ait ihtiyaçların ve hedeflerin tanımlanması, tasarımcıya, doğru bir yönlenme sağlamamaktadır.

2.2. Preiser (ed): Professional Practice in Facility Programming (1993)

Preiser, bir önceki derleme kitaptan bu yana, programlama disiplini çerçevesindeki

yenilikleri şöyle sıralamaktadır: Kullanım Sürecinde Değerlendirme'nin programlamanın tamamlayıcı bir parçası haline gelmesi, programlamanın mühendislikte kullanımının yaygınlaşması, belli tipte binalar için tasarım kılavuzları oluşturulmasında programlamanın öneminin artması, bilgisayar kullanımının artmasıyla programlama, değerlendirme ve işletmenin birbirine yakınlaşması, katılımcı programlamanın yaygınlık kazanması, adaptasyonla yeniden kullanımda programlamanın önem kazanması ve programlamaya emniyet, güvenlik, enerji kullanımı gibi yeni konuların katılması. [15] Tüm bu gelişmeler, programlamayı, işveren ve programlamacı ile sınırlı bir işbirliğinden çıkartmış ve tüm katılımcıların bulunduğu bir alan haline getirmiştir. Görüldüğü gibi, programlamanın etki ve kullanım alanı genişlemiştir. Kitap, bu genişlemiş kullanım alanına ilişkin örneklerin bir derlemesidir.

Hershberger, kitaba yazdığı *sonsözde*, kitapta yer alan örnekler bağlamında, programlamanın 1990'lı yıllarda ulaştığı özel konuma dair çıkarımlar yapmaktadır. [16] Preiser'den farklı olarak vurguladığı en önemli nokta, programlama ve tasarımın, bir bina oluşturmanın iki ayrı alt aşaması olmaması, tam tersi, artık bu iki alanının birbiri ile eş zamanlı olarak yürümesi gerekliliği görüşünün uygulamada yaygınlık kazanmış olmasıdır. Buna bağlı olarak, tasarımcı da programlamanın içinde, programlamacı da tasarımın içinde bulunacaktır. İkinci önemli bir nokta, programlamanın sonu olan bir eylem alanı olmayıp, kurum ve yapı bir arada var oldukça devam eden ve sürekli kendini yenilemesi gereken, devamlı ve dinamik bir eylem haline gelmesidir. Bu durumun önemli bir etkisi, sürekli bir programlamayı, kurum dışından uzman kişilerin değil, kurum içinden kişilerin yapması yani bina programlamanın, kurum içinde kendine bir alan bulmasıdır. Kurumun bir fabrika, bir yönetim merkezi, ya da bir hastane gibi değişimi ve sürekliliği gerektiren bir yapıda olması, bunun en önemli nedenidir.

Bu gelişmelerle programlama, kendine özgü işleyişi ve uzmanları olan bir ek hizmet alanı olmaktan çıkmış, kurumun, yapının ve yaşamın tam kalbinde yer alarak, onun organik bir parçası haline gelmiş olmaktadır. Dolayısıyla, programlama için hem tasarımla hem de yaşamla bütünleşmeye geçiş söz konusu olmaktadır. Bu da, mimarlık alanının belli bir parçasına değil bütününe yönelişinin bir ifadesidir. Temel paradigma, ihtiyaçlara en iyi cevap veren tasarımlar için veri oluşturmak gibi sınırlı bir alandan çıkıp, *“işverenin en yüksek amaçlarına, kurumun vazgeçilmez hedeflerine, çağdaş toplumun değerlerine, tasarımcının uygun formun damarlarda akan kan olduğu yolundaki inancına hizmet edebilen”* geniş ve etkili bir alana yayılmaktadır. Bu da, her bir yer ve kurumun bir diğerinden farklı işleyen bir organizma olarak algılanması gerekliliğini beraberinde getirmektedir. Artık programlamacı, adım adım (step-by-step) izlenen ve olguların/gerçekliklerin analitik bir yaklaşımla ele alınması yöntemiyle değil, yeni oluşum üzerine düşünce üreterek, onu anlamaya, kavramaya ve giderek de hissetmeye çalışarak çalışması gereken kişidir. Bu, programlama için önemli bir paradigma değişimidir (paradigm shift). Tamamen nesnellik üzerine kurulu teorik temelli bir model olma anlayışından,

özneliğin önceden kestirilemeyen ve sınıflandırılmayan tek defaya özgülüklerini de barındıran bir gerçekliğe geçiştir. Aslında bu değişim, mimarlık alanıyla tam bir örtüşme anlamına gelmektedir. Artık programlama bina elde etmek için değil, mimarlık için yapılmaktadır. Tek eksiklik, bu paradigma değişimindeki yeni durumu kendine temel alan, yeni bir programlama modelinin ortaya konmasıdır. Bu da bir sonraki aşamada gerçekleşmiştir.

2.3. Duerk: Architectural Programming (1993)

Programlama üzerine konu (issue) bazlı bir yaklaşım sunan Duerk’in kitabına önsöz yazan M. Brill, programlamayı “*bir çeşit tasarım araştırması; bilgi toplama, analiz ve öneri çevrenin tüm etkilerinin araştırıldığı verilerin düzenlenmesine yönelik disiplinli bir süreç*” [17], olarak tanımlamaktadır. Bu, kendinden öncekilerden uzaklaşmayan bir tanımdır. Ancak, bu tanımın ardından gelen programlamanın içeriğine yönelik diğer belirlemeler, öncüllerden çok, uygulamada kazanılan tecrübenin ve zamanın gereklerinin izlerini taşımaktadır. “*Programlama gerçekte insanların ve kurumların işlevsel, ekonomik, ruhsal ve sembolik olarak neye ihtiyaç duyduklarını araştırdığından, planlama ve tasarımın merkezine insan ihtiyaçlarını koyar. Bu durum kendi başına, yaşamlarımız için bir yer oluşturma (place-making) yönünde bir tavır, bir ideolojidir*” [17]. Bu ifade, artık programlamanın yaşam ve bina üretim süreçleri içinde kendine ait ve diğer tüm alanlarla bire bir ilişkili bir duruş edinmiş olduğunun ifadesidir. Ve bu duruşunda, bilimden çok insanla etkileşimdedir. Duerk’in yaklaşımı da tam bu etkileşim üzerine kuruludur.

Duerk, yaklaşımında fiziksel çevre-insan ilişkisinin elemanları olarak da yorumlanabilecek mahremiyet, emniyet, imge, fiziksel konfor, görsellik vb. konulara (issues) yönelik kavram (concept) geliştirme çabasıdadır. Geliştirilen kavramlar, programın maddeleridir. Yaklaşımında, öncelikle olgular (facts) gruplanır ve analiz edilir. Buradan, “*insanların neyin iyi neyin arzulanabilir olduğu hakkında karar vermede kullandığı ölçütler olan*” değerlere geçilir [4]. Hedeflenen kalite düzeyinin niteliklerini belirleyen hedeflerin (goals) belirlenmesinden sonra bu hedefler açık birer performans gerekliliği (performance requirement) ifadelerine dönüştürülür. Son aşama, tüm bu yapılanların tasarımcıya faydalı olacak kavramlara dönüştürülmesidir, zira tasarımcı için performans, yalın haliyle, göz ardı edilebilir bir konudur, fakat ideal organizasyonel ilişkileri gösteren diyagramlar, tasarımcının ilgisine ve kullanımına çok daha yakındır. Dolayısıyla, modelin kuruluşunda tasarımla doğrudan ilişkili olma, temel bir kriter olarak kendini var etmektedir.

“*Konu bazlı programlama, tekil bir projenin başarısında vazgeçilemez olan tasarım kararlarını net bir şekilde tanımlayarak, bir tasarım projesi için tek defaya özgü (unique) potansiyeli açığa çıkartmak için bir tekniktir*” [4]. Görüldüğü gibi Duerk, tek defaya özgülük ve değerler gibi mimarlığın temel kabullerini, kendi yaklaşımının merkezine yerleştirmektedir.

Programlama yaklaşımları alanına en önemli katkısı, tasarım alanıyla bütünleşmede, kavram olarak adlandırdığı ve *“biçim (boyut ve yön), malzeme, doku, renk ve yakınlıklar gibi mimarın kontrolü altında olan ideal bir ilişkiler düzeninin ifadesi”* [4] olarak tanımladığı, basit diyagramlar ve birkaç kelimedenden oluşan ve programın gerekliliklerinin nasıl yerine getirileceğine yönelik fikir verme amaçlı, bir ara alanı tanımlamasıdır. Tasarımla entegrasyon, bu ara alanda gerçekleşmektedir. Kavramların, eskizler ve sirkülasyon şemaları gibi ileri aşamalara götürülmesi, programlamanın tasarıma doğrudan dahil olmasını da beraberinde getirmektedir.

Kitabın *“uygulamalar”* başlığı altındaki ikinci bölümünde ise Duerk, bilimsel metot ve bilimsel metodun bina programlamayla çakışan araştırma alanları, metotları ve tekniklerine yer vermektedir. Çevre-davranış araştırmalarını ve değerlendirme yaklaşımlarını da bu kapsamda ele almaktadır. Bu durum, programlama disiplininin ilk örneklerini oluşturan yaklaşımların etkisinin halen sürdüğünün bir göstergesi gibi görünmektedir. Buna karşılık, bilimsel metodun yalnızca araştırma teknikleri ile ilişkili kılınması, programlamanın tanım, içerik ve etki/kapsama alanı ile ilişkilendirilmemiş olması, ilk örneklerden bilinçli bir kopuşu işaret etmektedir. Bilimsellik, artık bir amaç değil, olumlu çevrelerin yaratılmasında gerekli araştırma stratejilerinin temelini oluşturan bir araçtır.

3. Adım: Tasarım, Mimarlık ve Yaşamla Bütünleşme

1990’lı yılların sonunda bina programlamada geline en son noktada artık disiplin, tasarımla, yaşamla ve yaşamı paylaşan tüm aktörlerle (kullanıcı, işveren, müteahhit, vb) sıkı bir ilişki içindedir. Her bir aktörün/grubun kendi değerlerini ortama katması, sonuç kararın da bilimsel akıl yürütme teknikleri ile değil, aktörler arası iletişimle varılan ve değişkenliği en baştan kabul edilen bir durum olduğu kabulünü beraberinde getirmiştir. Bu durumda programlamacı da, düşünceleri toparlayan ve uzlaşmayı sağlayan kişi olur, yani bir anlamda bir iletişim uzmanlığı görevini yürütür. Artık programlamacı, bir psikolog, bir antropolog, bir yönetim danışmanı, bir sanatçı ve bir matematikçidir. Bağlamı ve bağlam içinde bireylerin konumlarını ve değerlerini ele geçirmeye çalışan iyi bir dinleyicidir [18]. Kişileri/grupları doğru anlamak, diğerlerine doğru aktarmak ve her bir değer ifadesinden yararlanarak olabilecek en uzlaştırıcı/en ortak çözümü bir tasarım problemi dili çerçevesi içinde ortaya koymak. Dolayısıyla, programlamacının rolü, bilimsel olduğu iddia edilen katı bir çerçeveden çıkarak, yaşamsal olan dinamik ve çok aktörlü bir ortama geçmiştir. Öyle ki, artık bir bina programı oluşturmak için kurum dışında, uzmanlaşmış bir programlamacı aranmamaktadır. Kurumlar kendi içlerinde, bina programını, değişen yaşam koşullarına, istek ve ihtiyaçlara adapte etmekle görevli kişileri barındırmayı kendileri için bir artı değer olarak görmektedirler. Buradan hareketle, bina programlamanın, uzmanlaşmış programlamacıları da aşarak, kurumsal değerlerle birlikte evrilen ve kurumun organik bir parçası olarak kabul edilen görece daha yaşamsal bir role kavuştuğunu iddia etmek yanlış olmayacaktır. Bu yeni rolde, başlıca iki tutum mevcuttur: (i) Bilgi toplama ve karar verme

mekanizmalarında programlamacının rolünü olabildiğince nesnel tutmayı amaçlayan, bunun için de onun analizci ve soyutlayıcı düşünce yapısını ön planda tutan tutum ve (ii) Aktörlere ait değerlere hassas yaklaşmayı esas alan tutum. Her iki yaklaşım da insan-çevre ilişkisinin olumlu gerçekleşeceği mekanların tasarımına, doğru tanımlanmış problemlerle katkı koymayı amaçlamaktadır. Her iki duruma it örnek kitapların açılımları aşağıda verilmiştir.

3.1. Cherry: Programming for Design (1999)

Orta ölçekli bir proje için gerekli programı oluşturmada kullanılacak düşünce biçimleri, araştırma teknikleri ve programlama stratejilerini, disiplinle ilk defa tanışmakta olan öğrencilere aktarma amaçlı yazılmış olan kitapta programlama, öncüllerinden farklı olarak “*elde edilebilen tüm bilgiden değerli olan bilgiyi ayırt etmek, verili tüm olası hükümlerden en önemlilerini seçebilmek, sinyali sesten ayırmak*” olarak tanımlanmaktadır [19]. Tanımda odaklanılan nokta, bir metot değil bir düşünce biçimidir.

Bunun içindir ki kitap, düşünme üzerine düşünmeyle (think about thinking) başlar. Düşüncenin analiz, sentez, tümdengelim, tümevarım, lineer ve holistik düşünce, Doğu felsefesi ve fuzzy logic hallerinin tanımlanmasından ve programlamayla ilişkilerinin irdelenmesinden sonra Cherry'nin vardığı yargı şudur ; “*Mimari programlamacılar, çok yönlü (versatile) düşünürler olmalıdırlar –kendi düşünce süreçlerini duruma göre biçimlendirebilmelidirler (tailor)*” [19]. Bir sonraki bölümde, işverene üzerine odaklanılır. Cherry, işvereni, bir tesisin yapılmasıyla etkilenen tüm insanlar dolayısıyla da oluşturulmasında etkisi olması gereken insanlar topluluğu olarak tanımlamaktadır. Öncüllerin de genişçe yer verdiği bilgi toplama yöntemleri ile işverene nasıl ulaşılabileceğini aktardıktan sonra Cherry, kendi programlama yaklaşımına yer verir.

Programlama yaklaşımında en önemli özelliği, sistemli olmasıdır. Bir programlamacının karşısına çıkabilecek tüm problem alanları sistemli bir şekilde sınıflanmış ve açıklanmıştır. Kişinin yapması gereken tek şey, eline gelen bilgiyi, ilgili olduğu başlığın altına yerleştirmektir. Örneğin; yer analizi için, yerin bağlamı, yasal tanımlamalar, zonlamalar, trafik, servisler, topografya, manzara, yapılı çevre, toprak koşulları vd. alt başlıkları kullanır. Bu tutum, programlamanın başlangıçtaki sınıflandırıcı düşünce biçimine paralellikler taşımakla birlikte, geçen zamanın ve edinilen deneyimlerin de zenginleştirdiği daha incelmış/detaylanmış bir bakış açısının varlığı kendini hissettirmektedir.

Cherry'nin programatik stratejiler olarak adlandırdığı, esneklik, merkezilik, akış, enerji korunumu, öncelikler, ulaşım vb. kavramları, aslında Pena'dan bu yana var olan -Duerk'in de programlamada konular (issues) olarak adlandırdığı- projenin kalitesini işverenin memnuniyeti bağlamında tanımlayan, programlaması yapılan tesis ile ilgili düşünceleri soyut biçimde ifade eden ve bu konuda programlamacının

da yaratıcı olmasını gerektiren ifadelerdir. Tasarımcı, bu soyut ifadeler üzerinden somut biçimlere varacaktır. Benzer şekilde, *özet ifadeler* (summary statements) olarak adlandırılan ve tüm araştırma bulgularının tasarımcıya aktarılacak bir ortama dönüştürülmesini ifade eden sentez aşaması da, Pena’dan bu yana var olan programlamacı-tasarımcı ayrımının kabulüne dayanan bir devamlılıktır. Dolayısıyla Cherry’nin öncüllerinden kopmadığını fakat onları, yeni zamanın ve deneyimlerin de katkısıyla zenginleştirerek bugüne uyarladığını iddia etmek yanlış olmayacaktır. Tekrar belirtmek gerekir ki, bu uyarlamada *insan ve yaşam* merkezdedir.

Yaklaşımını açıklamaya başlarken Cherry, programlamayı bu kitaptan öğrenecek olanlara bir öneride bulunur; aslında kitapta programlama adımları lineer bir sırayla aktarılmaktadır ancak öğrenen kişi kendisine gerçek ya da gerçeğe yakın bir programlama problemi bulmalı ve bu sürecin aslında lineer değil, holistik olduğunu kendisi keşfetmelidir. Cherry’nin kendi modelini açıklamasından önce işvereni anlamaya çalışması anlamlıdır, sürece ya da bilgiye değil, insana ve onun değerlerine öncelik verildiğinin, diğer süreçlerin işverenden alınacak ipuçlarıyla yönetilmesi gerektiğinin bir göstergesidir. Bu tip bir süreçte, altta yatan temel paradigma, programlamanın, “*bir projenin biricikliğinin (uniqueness) tanımlanması*” [19] olarak görülmesidir. Dolayısıyla, artık standartlaştırıcı, tiplleştirici, genelleyici programlama anlayışı, yerini mimari tasarımın da doğasında var olan, *tek defaya özgü* anlayışına bırakmıştır.

Profesyonel ya da kuramsal alana değil eğitim alanına yönelik olan bu kitapta Cherry, programlama disiplininin geçmiş kabul ve sınıflamalarından tamamen vazgeçmemekte, onlara insanın, çevre içinde gerçekleşen yaşamın ve mimarlık alanının doğasını katarak yeni bir boyut eklemektedir. Bu işlemde Cherry, mevcut sistematik bakışı bozmamakta fakat onu, olası her durumu içerebilmesi amacıyla derinleştirmektedir.

3.2. Hershberger: Architectural Programming and Predesign Manager (1999)

Cherry’nin kitabı gibi Hershberger’in eseri de eğitim amaçlıdır. Dolayısıyla, programlamanın tanımı ve bugüne kadarki programlama yaklaşımlarının dökümlenmesi ile başlar. Kendi yaklaşımını öncüllerin üzerine kurar. Programlama yaklaşımlarını açıklamada en ağırlıklı bölümü değer bazlı yaklaşıma (value-based approach) ayırır. Bu yaklaşım, Hershberger’in bir önceki aşamada yer alan değerler üzerine kurulu söyleminin bir devamı ve bir yaklaşım haline getirilmiş versiyonudur. Aslında bu yaklaşımın bir programlama yaklaşımı mı yoksa bir tasarım yaklaşımı mı olduğunu anlamak güçtür. Zira, bu yaklaşıma verilen örnekler, diğer yaklaşımların dayandırıldığı gibi, teorik örnekler olmayıp F.L. Wright, L.Kahn, Straub ve Bruder’in tasarım yaklaşımlarına ve Hershberger’in kendi programlama ve tasarım birlikteliğine dayalı uygulamalarına dayanır. Örneklerin ortak noktası şudur; mimar tasarımı gerçekleştirmeden önce, işveren ile birlikte yaşar, konuşur, onun şiirsel ya da pragmatik tüm arzularını ele geçirir. Tasarıma bu

ifadelerin kendisinde uyandırdığı ve tasarımın biricik/kendine özgü bir eser olmasını sağlayan özel problemlerin çözümü ile başlar. Böylece tasarım, en sonunda da yapı, işverenin/kullanıcının gerçekte hayal ettiği, arzuladığı ve ihtiyaç duyduğu bir ortam haline gelir, o işverenin/kullanıcının olur [20]. Görüldüğü gibi, artık programlama yaklaşımı olarak kendine özgü bir alan kalmamış, programlama tasarım olarak tanımlanmıştır. Straub’a dayandırılan “*program tasarımıdır*” (The program is the design) ilkesi ise bu yaklaşımın kısa/somut ifadesidir.

Hershberger için bu tip bir yaklaşım diğer tüm yaklaşımlardan üstündür, çünkü en önemli ve iyi unsurlara kendini açar ve en kötü hatalardan korunmanın en etkili yoludur. Bu iddiasını Hershberger şöyle temellendirir: (i) Değer bazlı programlama, tasarım probleminin esas doğasını araştırır, işverenin/kullanıcının güçlü bir şekilde tutunduğu değerleri ve hedefleri ele geçirir, böylece programlamanın diğer eylemlerinin –ve de tasarımın- bu değerler ve hedefler etrafında biçimlenmesine neden olur. (ii) Değer bazlı programlama, programlama süreci içinde elde edilecek işverene ait ve diğer bilgilerin güvenilir ve geçerli olması için bilgi-temelli programlama yaklaşımının sistematik sürçlerini de kullanır. Bu süreçler, önceki yaklaşımlarda da aktarılan, anket, gözlem vb. bilimsel bilgi üretme metodlarıdır. Dolayısıyla, temel değerler ve hedefler ele geçirildikten sonra, bunlar ve diğer alanlar için sistematik araştırmaya girişmek, başlangıçta ne aradığını bilmeden yola çıkan bir sistemli araştırmadan daha yararlıdır. (iii) Değer bazlı programlama, Pena’nın da yer verdiği, uzlaşma-temelli metodu (agreement-based method) kullanır. Buna göre, tüm aktörler kendilerini ifade ederler. Çatışan noktalarda programlamacı tarafından uzlaştırılırlar. Pena’nın yaklaşımından farklı olarak, değer kategorileri daha zengindir, öncelik sıralaması her projeye göre farklılaşabilir esnekliktedir; hem program hem de tasarım fikirlerini birlikte içerir, böylece tasarımcıya daha geniş bir fikir yelpazesi sunar; tasarımcı programlama sürecinin içinde olsun ya da olmasın, en önemli tasarım konuları üzerine bir görüş geliştirebilmesi, konuyu özünden kavrayabilmesi esastır; program, tasarım geliştirme bilgilerini de içerir, her bir mekanı oluşturacak özel koşullar (boyut, oran, eşya, kullanım sıklığı, kullanıcı sayısı vb.) için tanımlar ortaya koyar, böylece tasarımcının mekanları özgün gerçekleriyle hissedebilmesi sağlanır. Tüm bunlarda hedef “*iyi/kaliteli mimarlık*”tır.

Hershberger yaklaşımında, öncelikle *Değerler ve Mimarlık* başlığı altında, mimarlığın barındırdığı değerleri listeler, içeriklerini açıklar. Ve bir sonraki aşamada, değerleri program konularına (issues) dönüştürür. Bu konular, tasarımcının problem çözümünde üzerinde durması gereken temel noktalar. Yaklaşımda, kalıcı (korunma, iyi hayat ve sanat) ve çağdaş (bağlam, çevre, iklim, teknoloji, sosyal yapı, gelenek vb) kategorilerinde ele alınan değerler, Pena’nın dört konu başlığına karşın (işlev, biçim, ekonomi ve zaman) burada sekiz konu başlığı altında (insani, çevresel, kültürel, teknolojik, geçici, ekonomik, estetik, ve güvenlik) ele alınır ve beş aşamada (değerler, hedefler, olgular, ihtiyaçlar ve fikirler) geliştirilir. Tüm süreç çalışma oturumları (work sessions) ile işverene/kullanıcıya aktarılır ve onaya sunulur. Tüm

çalışma, sistemli bir programla belgelenir. Aslında, değerleri, uzlaşmacı bir yaklaşımla ele geçirdikten sonra, bunları konulara dönüştürmek ve üzerlerinde daha derin araştırmalara girişmek, programlamanın en baştan beri var olan sistematik yaklaşımının bir devamıdır. Burada sistematiklik, yön veren bir dümen ile belli bir rota kazanmıştır. Bu dümen değerlerdir, rota ise kaliteli/iyi mimarlıktır.

Hershberger’in programlama süreçleri ve bilgi toplama teknikleri ile ilgili dökümü, öncüllerinin bir derlemesidir. Özellikle bilgi toplama tekniklerinde öncüllere göre zenginleşen ve daha kategorikleşen bir içerik göze çarpar. Kitabın, uygulamacı, programlamacı ve eğitmen bir kişi tarafından yazılmış olması bu sistemliliğin ve zenginliğin temel nedenidir.

Kitapta son olarak, programlamaya katkısı açısından değerlendirme problemi ve uygulama alanları üzerinde durulur. Değerlendirmenin, mimarlık mesleğinin sanatsal ve bilimsel içeriğinin zenginleşmesi için gerekliliği vurgulanır. Buradaki değerlendirme, (i) programın, (ii) tasarımın her bir aşamasının, (iii) tamamlanmış inşaatın, (iv) kullanımın hemen öncesindeki yapı durumunun ve (v) kullanım sürecindeki durumun değerlendirilmesi olarak, bina üretiminin her durak noktasında gerçekleştirilmesi gereken bir eylem olarak ele alınmaktadır. Bu yaklaşım da, baştan beri var olan bilimsel düşünüşün bir yansımasıdır.

Görülmektedir ki, Hershberger, programlamaya değerlerin ele geçirilmesiyle başlamayı tercih etmekte ve bu yolla tüm bir sürecin bu değerler üzerinde odaklanmasını garanti altına almayı hedeflemektedir. Doğaldır ki bu durum, insanın kendi yaşayacağı çevre üzerindeki söz hakkını sonuna kadar kullanması gerekliliğine dayanır. Yani kararlar ve tercihler, programlamacı ya da tasarımcı gibi kişilerin kabullerine ve ön yargılarına bırakılmaz. Her bir problem ve ona yönelik karar, belli bir değer sisteminin parçası olarak ele alınır. Bu değerler sisteminin keşfi ve açıkça ifade edilmesi, üretim sürecini, ürün kalitesini ve bunlara bağlı olarak da yaşam kalitesini yükseltir. Zaten, iyi mimarlığın tanımı da tam bu noktada temellenmektedir.

SONUÇ

Çalışma boyunca, bilimsellikten yaşamsallığa olan serüveni örnek eserler aracılığı ile açıklanmaya çalışılan bina programlama -asında bugün ulaştığı noktadaki ismiyle Mimari Programlama-, yaklaşık 40 yılı kapsayan bir zaman dilimi içinde kaliteli mimarlık arayışının ayrılmaz, olmazsa olmaz bir parçası haline gelmiştir. Bu değişimde, bir iletişim ortamı olma durumu her zaman varlığını korumuş olmakla birlikte, disiplinin kuruluş aşamalarında bu rol görece örtüktür. Bugün geldiği noktada ise programlama, yoğun bir iletişim ortamına dönüşmüştür. Bu iletişimde, değerleri açığa çıkarma ve farklı değerleri barındıran kişi ve gruplar arasında uzlaşma sağlama gibi etkin iletişimi gerekli kılan taktikler ön plana çıkmıştır.

Dolayısıyla artık programlama, tüm katılımcıların bulunduęu ortak bir mekan, bir arenadır.

Sistemlilik, programlamanın ilk ortaya çıktıęı zamanlardan itibaren varlığını ve gücünü hissettiren bir özellik olarak korunagelmiştir. Bugün gelinen noktada, sistem ve kategorizasyon, başlangıç aşamalarına göre çok daha zengin ve derindir. Artık, her olası durumda, içinden gerekli olanların seçilebildięi bir katalog haline gelmiş olan programlama konuları ve arařtırma metotları, yeni programlar için zengin bir veri kaynağıdır.

Program, artık, tasarımdan önce tamamlanan, bir daha deęişmeyen, donmuş ve statik bir tanım olmaktan çıkmış, zamanla yenilenen ve tasarımla, tasarımın kendisine ihtiyaç duyduęu her aşamada ve alanda etkileşebilen bir alan haline gelmiştir. Hershberger, deęerleri keşfetmeyi programlama sürecinin en başına almakla, aslında tasarıma ve programlamaya eş zamanlı bir başlangıcı da önermiş olmaktadır. Başlangıçta kendisini yoğun bir biçimde hissettiren programlamacı ve tasarımcı ayrımı, bugün ayrım olsun ya da olmasın her iki aktör grubunun da deęerler üzerine odaklanmasıyla aşılmıştır. Dolayısıyla, programlama yaşam kadar deęişken ve aktif bir alan haline gelmiştir. Çünkü, görevi onu izlemek ve desteklemektir. Söz konusu durum, programlamanın ayrı bir disiplin olarak kendini ortaya koymasından çok önceleri için de geçerlidir. Deneme-yanılma yoluyla elde edilen binalar içinde kültürel ve fiziksel kořullara en uygun olanların benimsenmesi ve tekrarlanması [1] programın yaşamla olan baęının her dönem geçerli bir baę olduęunun kanıtıdır. Programlamanın bugün geldięi nokta, gerçekte, bir disiplin haline gelmeden önceki durum ile de özdeşlik göstermektedir.

Programlama artık, tasarım bürolarının bir ön ve ek hizmeti olmaktan çıkarak, her kurumun kendi içinde gerçekleřtirmesi gereken ve kurumsallığın doğal bir parçası olan bir süreçtir. Yaşamın, ihtiyaçların ve mekanın dönüşümü, kurumların evrilebilmesi için, yakından izlenmesi ve mutlaka pozitif adaptasyonlarla sonuçlanması gereken bir durumdur.

Tüm bu deęişimlerle programlama, tüm yapı tipleri için standart bir süreç olma özelliğini taşıyan, ancak amacı her bir yapının kendine özgü, biricik karakterini tanımlamak olan kısaca, standart süreçlerle, asla standartlaştırılmayan tek defaya özgümlükleri yaratmaya çalıřan bir anlayışın en önemli parçası haline gelmiştir. Bu durumda programlama -Brill’in de işaret ettięi gibi- bir ideolojidir, bir tavidir, bir duruştur. Tip mekânlarda, tip yaşamlar ve tip problemler yerine, kullanıcısının ihtiyaç ve isteklerine özgü, olanakların ve kısıtların da belirleyiciliğinde, tek defalık çözümlerin üretilmesini hedeflemektedir. Yöntem “1” sonuç/fikir/keşif, “n” sayıdadır. Yalnızca bu tip bir anlayışla üretilen fikirler, bir tasarımı ve sonuç ürünü, insanları mutlu edebilen bir sonuca ulařtırabilecek güçtedir. Bu güç, özellikle kullanıcının mekânı üzerindeki belirleyiciliğinden ve onu istedięi zaman istedięi şekilde yeniden kurgulayabilme özgürlüğünden kaynaklanmaktadır. Başlangıçta

sözü edilen belirleyicilik, dönüştürücülük ve simgesellik değerleri insan-mekan ilişkisinde kendini bu şekilde var eder. Bu anlayışla, insan ve tasarım ürünü olan fizik çevre, hiç olmadığı kadar bütünlüşir ve mekan, içinde barındırdığı insanların organik bir uzantısı olur. Sonuçta **mekan, el ve ayak gibi işleyebilen, zihin ile birlikte düşleyebilen ve insanın varoluşsal serüvenini zenginleştiren bir varlık haline gelir.** Konunun bundan sonraki şiirsel (poetic) açılımı, bir başka çalışmanın içeriğidir.

Kısaca, Mimari Programlamanın bugünkü konumunu belirleyen temel değerler bilim değil “*yaşam*”, sistem değil “*insan*”; temel eylemi ise “*analiz etmek*” değil “*anlamak/kavramak/düşünmek/tasavvur etmek (to conceive)*” olarak özetlenebilir.

KAYNAKLAR

1. Onat, E., **Mekansal Organizasyonlarda İhtiyaç Programlaması**, Teknik Kitabevi, Ankara, 1990
2. Preiser, W.F.E., “Design Innovation and the Challenge of Change”, **Design Intervention ; Toward a More Humane Architecture**, Van Nostrand Reinhold Company, USA, 1991
3. Merriam-Webster OnLine Dictionary,
<http://www.m-w.com/cgi-bin/dictionary?book=Dictionary&va=program>
4. Duerk, D., **Architectural Programming**, Van Nostrand Reinhold Company, New York, USA, 1993
5. Sanoff, H., Integrating Programming, Evaluation and Participation in Design, Avebury, England, 1992
6. Brown, J.M., “Programming, Architectural”, **Encyclopedia of Architecture, Design and Construction**, J.A.Wilkes (ed), AIA, John Wiley & Sons, USA, 1989
7. Hershberger, R., “Values: A Theoretical Foundation for Architectural Programming”, **Programming the Built Environment**, Preiser, W.F.E. (ed) Van Nostrand Reinhold Company, New York, USA, 1985
8. Pena, W., Parshall, S., Kelly, K., **Problem Seeking, An Architectural Programming Primer**, AIA Press , USA, 1977
9. Sanoff, H., **Methods of Architectural Programming**, Dowden, Hutchinson and Ross, Inc., USA, 1977
10. Preiser, W.F.E., (ed), **Programming the Built Environment**, Van Nostrand Reinhold Company, New York, USA, 1985
11. Preiser, W.F.E., (ed) **Professional Practice In Facility Programming**, Van Nostrand Reinhold Company, New York, USA, 1993
12. Preiser, W.F.E., “Introduction”, **Programming the Built Environment**, Preiser, W.F.E. (ed), Van Nostrand Reinhold Company, New York, USA, 1985

13. Hershberger, R.G., “Values: A Theoretical Foundation for Architectural Programming”, **Programming the Built Environment**, Preiser, W.F.E. (ed), Van Nostrand Reinhold Company, New York, USA, 1985
14. Silverstein, M., Jacobson, M, “Restructuring the Hidden Program: Toward an Architecture of Social Change”, **Programming the Built Environment**, Preiser, W.F.E. (ed), Van Nostrand Reinhold Company, New York, USA, 1985
15. Preiser, W.F.E., “Epilogue”, **Professional Practice In Facility Programming**, Preiser, W.F.E. (ed), Van Nostrand Reinhold Company, New York, USA, 1993
16. Hershberger, R., “Recent Development in Facility Programming”, **Professional Practice In Facility Programming**, Preiser, W.F.E. (ed), Van Nostrand Reinhold Company, New York, USA, 1993
17. Brill, M., “Foreword”, **Architectural Programming**, Duerk, D., Van Nostrand Reinhold Company, New York, USA, 1993
18. Pena, W., 1999, “Foreword”, **Programming for Design, From Theory to Practice**, Cherry, E., John Wiley & Sons Inc., New York, USA, 1999
19. Cherry, E., **Programming for Design, From Theory to Practice**, John Wiley & Sons Inc., New York, USA, 1999
20. Hershberger, R., **Architectural Programming and Predesign Manager**, McGraw-Hill, New York, USA, 1999